



Universidad  
Carlos III de Madrid

## **TRABAJO FIN DE GRADO**

**Título: Diseño e implementación de una interfaz  
de domótica asistencial basado en  
realidad aumentada**

**Autor: Santiago de Arriba Cortijo**

**Titulación: Grado en Ingeniería Informática**

**Tutor: Javier Fernández Muñoz**

**27 de septiembre de 2015**



## ÍNDICE DE CONTENIDO

1.	Introducción .....	10
1.1	Motivación .....	10
1.2	Objetivos .....	10
1.3	Contexto socio-cultural y económico .....	10
1.4	Estructura del documento .....	11
2.	Estado del arte .....	12
2.1	Sistemas de domótica .....	12
2.1.1	Introducción .....	12
2.1.2	Innovaciones .....	12
2.1.3	Arquitectura .....	14
2.1.4	Normativa y legislación .....	15
2.1.4.1	Normas Técnicas .....	15
2.1.4.2	Reglamentación de las instalaciones domóticas .....	17
2.2	Sistemas móviles .....	19
2.2.1	Introducción .....	19
2.2.2	Dispositivos móviles .....	19
2.2.3	Aplicaciones móviles .....	20
2.2.4	Sistemas operativos .....	20
2.3	Android .....	23
2.3.1	Introducción .....	23
2.3.2	Arquitectura de Android .....	24
2.3.3	Versiones de Android .....	27
2.3.4	Forks de Android .....	28
2.4	Códigos QR .....	30
2.4.1	Información general .....	30
2.4.2	Ventajas .....	30
2.4.3	Ejemplos de uso .....	31
2.4.3.1	Ampliación de información .....	31
2.4.3.2	Programa piloto 'Código QR' para enfermos de Alzheimer .....	32
2.4.4	Librería ZXing .....	32
2.5	Realidad Aumentada .....	33
2.5.1	ARToolKit .....	33
2.5.1.1	Proyecto AndAR .....	34
2.5.2	OpenCV .....	36

3.	Análisis del sistema .....	37
3.1	Identificación de escenarios y usuarios .....	37
3.2	Requisitos de Usuario.....	37
3.2.1	Prototipo de requisito de usuario .....	37
3.2.2	Requisitos de capacidad.....	38
3.2.3	Requisitos de restricción.....	41
3.3	Requisitos de software .....	43
3.3.1	Prototipo de requisito .....	43
3.3.2	Requisitos funcionales.....	44
3.3.3	Requisitos no funcionales .....	47
3.3.4	Matriz de trazabilidad de requisitos .....	50
3.3.5	Casos de uso.....	51
4.	Diseño del Sistema.....	55
4.1	Introducción .....	55
4.2	Opciones de diseño .....	55
4.2.1	Representación de información .....	55
4.2.1.1	ARToolkit .....	55
4.2.1.2	OpenCV .....	57
4.2.1.3	Solución adoptada .....	58
4.2.2	Reconocimiento de voz .....	59
4.3	Arquitectura del Sistema.....	60
4.4	Diseño UML del sistema.....	61
4.5	Interfaz del sistema.....	62
4.5.1	Diagrama de navegación.....	64
5.	Implementación del Sistema.....	65
5.1	Introducción .....	65
5.2	Herramienta de desarrollo del proyecto.....	65
5.3	Otros recursos .....	66
5.3.1	OpenCV.....	66
5.3.2	ZXing .....	67
5.3.3	SpeechRecognizer .....	67
5.3.4	Text To Speech .....	67
5.4	Estructura del proyecto.....	68
6.	Pruebas del Sistema.....	69
6.1	Introducción .....	69
6.2	Procedimiento de pruebas.....	69

7.	Gestión del Proyecto .....	77
7.1	Introducción .....	77
7.2	Metodología del proyecto .....	77
7.3	Definición de roles. ....	77
7.3.1	Propietario del producto.....	77
7.3.2	Equipo de desarrollo.....	78
7.4	Planificación inicial.....	78
7.4.1	Estimación y planificación de las tareas .....	78
7.4.2	Diagrama de Gantt .....	81
7.5	Presupuesto del proyecto .....	82
7.6	Planificación final .....	83
8.	Conclusiones .....	85
8.1	Conclusiones del proyecto.....	85
8.2	Conclusiones personales.....	85
8.3	Trabajos futuros.....	86
ANEXO I – Manual usuario de la aplicación .....		87
ANEXO II – Manual de creación de marcadores .....		90
ANEXO III– Acrónimos, definiciones y referencias. ....		92
Acrónimos.....		92
Definiciones .....		93
Referencias.....		94

## ÍNDICE DE TABLAS

Tabla 1: Principales organismos por campo de aplicación y ámbito geográfico .....	15
Tabla 2: Cuotas de mercado de SSOO móviles .....	22
Tabla 3: Relación de nombres de versiones Android .....	27
Tabla 4: Distribución de versiones Android en el año 2015 .....	28
Tabla 5: Tabla de tolerancia a errores QR.....	30
Tabla 6: Formatos compatibles con la librería ZXing.....	32
Tabla 7: Prototipo de requisito .....	38
Tabla 8: Requisito de Usuario de Capacidad 1. Mostrar información sobre los dispositivos.....	38
Tabla 9: Requisito de Usuario de Capacidad 2 - Detectar comandos de voz .....	39
Tabla 10: Requisito de Usuario de Capacidad 3 - Leer etiquetas.....	39
Tabla 11: Requisito de Usuario de Capacidad 4 - Mostrar información útil .....	39
Tabla 12: Requisito de Usuario de Capacidad 5 - Obtener información del servidor .....	39
Tabla 13: Requisito de Usuario de Capacidad 6- Menú de ajustes .....	39
Tabla 14: Requisito de Usuario de Capacidad 7- Dirección del servidor editable .....	40
Tabla 15: Requisito de Usuario de Capacidad 7- Formato de información editable.....	40
Tabla 16: Requisito de Usuario de Capacidad 9- Zoom de video.....	40
Tabla 17: Requisito de Usuario de Restricción 1 - Funcionamiento con internet.....	41
Tabla 18: Requisito de Usuario de Restricción 2 - Funcionamiento en Android .....	41
Tabla 19: Requisito de Usuario de Restricción 3 - Comunicación con servidor local.....	41
Tabla 20: Requisito de Usuario de Restricción 4 – Uso de cámara.....	41
Tabla 21: Requisito de Usuario de Restricción 5- Utilización de micrófono .....	42
Tabla 22: Requisito de Usuario de Restricción 6 - Uso de librerías estándar.....	42
Tabla 23: Requisito de Usuario de Restricción 7 - Uso sin contacto .....	42
Tabla 24: Requisito de Usuario de Restricción 8 - Uso sin datos .....	42
Tabla 25: Requisito funcional 1 - Mostrar información sobre la captura .....	44
Tabla 26: Requisito funcional 2 - Interactuar con comandos de voz.....	44
Tabla 27: Requisito funcional 3 - Reconocer etiquetas.....	44
Tabla 28: Requisito funcional 4 - Mostrar ayudas visuales .....	45
Tabla 29: Requisito funcional 5 - Obtener comandos del servidor.....	45
Tabla 30: Requisito funcional 6- Definir ajustes.....	45
Tabla 31: Requisito funcional 7- Ajustar dirección del servidor.....	46
Tabla 32: Requisito funcional 8- Ajustar formato de información.....	46
Tabla 33: Requisito funcional 9- Aumentar zoom de video por voz .....	46
Tabla 34: Requisito funcional 10- Disminuir zoom de video por voz.....	46

Tabla 35: Requisito no funcional 1 - Funcionamiento con internet .....	47
Tabla 36: Requisito no funcional 2 - Funcionamiento en Android.....	47
Tabla 37: Requisito no funcional 3 - Funcionamiento en red local.....	47
Tabla 38: Requisito no funcional 4 - Funcionamiento con cámara .....	48
Tabla 39: Requisito no funcional 5- Funcionamiento con micrófono.....	48
Tabla 40: Requisito no funcional 6 - Funcionamiento con API de Google .....	48
Tabla 41: Requisito no funcional 7 - Funcionamiento sin contacto .....	49
Tabla 42: Requisito no funcional 10 - Funcionamiento sin datos.....	49
Tabla 43: Matriz de trazabilidad de requisitos.....	50
Tabla 44: Caso de uso 1 - Aumentar zoom .....	51
Tabla 45: Caso de uso 2 - Disminuir zoom .....	52
Tabla 46: Caso de uso 4 – Leer etiqueta .....	52
Tabla 47: Caso de uso 5 – Desactivar sonido .....	52
Tabla 48: Caso de uso 6 – Activar sonido .....	53
Tabla 49: Caso de uso 7 – Operar comando .....	53
Tabla 50: Caso de uso 8 – Visualizar ajustes .....	53
Tabla 51: Caso de uso 9 – Modificar ajustes .....	54
Tabla 52: Caso de uso 10 – Descartar cambios .....	54
Tabla 53: Caso de uso 11 – Salir.....	54
Tabla 54: Valoración de prototipos .....	59
Tabla 55: Prueba funcional P – 01.....	69
Tabla 56: Prueba funcional P – 02.....	69
Tabla 57: Prueba funcional P – 03.....	70
Tabla 58: Prueba funcional P – 04.....	70
Tabla 59: Prueba funcional P – 05.....	70
Tabla 60: Prueba funcional P – 06.....	71
Tabla 61: Prueba funcional P – 07.....	71
Tabla 62: Prueba funcional P – 08.....	71
Tabla 63: Prueba funcional P – 09.....	72
Tabla 64: Prueba funcional P – 10.....	72
Tabla 65: Prueba funcional P – 10.....	72
Tabla 66: Prueba funcional P – 11.....	73
Tabla 67: Prueba funcional P – 12.....	73
Tabla 68: Prueba funcional P – 13.....	73
Tabla 69: Prueba funcional P – 14.....	74
Tabla 70: Prueba funcional P – 15.....	74
Tabla 71: Prueba funcional P – 16.....	74
Tabla 72: Prueba funcional P – 17.....	75
Tabla 73: Prueba funcional P – 18.....	75
Tabla 74: Prueba funcional P – 19.....	76
Tabla 75: Prueba funcional P – 20.....	76
Tabla 76: Prueba funcional P – 19.....	76
Tabla 77: Tareas del proyecto .....	78
Tabla 78: Tareas de implementación y pruebas .....	79
Tabla 79: Planificación y estimación del proyecto .....	80
Tabla 80: Planificación de implementación y pruebas .....	80
Tabla 81: Presupuesto del proyecto .....	83
Tabla 82: Comparativa de fechas estimadas y reales .....	84

Tabla 83: Comparativa de horas estimadas y reales ..... 84



## ÍNDICE DE ILUSTRACIONES

Ilustración 1: Arquitectura de sistema centralizado.....	14
Ilustración 2: Arquitectura de sistema descentralizado .....	14
Ilustración 3: Arquitectura de sistema distribuido.....	14
Ilustración 4: Grafica de cuota de mercado .....	22
Ilustración 5: Implantación de Android en otros dispositivos.....	23
Ilustración 6: Evolución de los dispositivos con SO Android.....	24
Ilustración 7: Evolución de Android Market.....	24
Ilustración 8: Capas de la arquitectura Android .....	25
Ilustración 9: Estructura de arquitectura Android .....	26
Ilustración 10: Distribución de versiones Android en el año 2015.....	28
Ilustración 11: Código QR con capacidad de 300 caracteres alfanuméricos .....	30
Ilustración 12: Sistema de orientación del código QR .....	31
Ilustración 13: Comparativa de versiones y tamaño .....	31
Ilustración 14: Programa piloto para enfermos de Alzheimer .....	32
Ilustración 15: Logo de ZXing .....	33
Ilustración 16: Marcador de ARToolkit.....	34
Ilustración 17: Marcador original, con matriz de 64x64 y de 16x16 .....	34
Ilustración 18: Ejemplo de realidad aumentada del proyecto AndAR ....	35
Ilustración 19: Ejemplo de realidad aumentada del proyecto AndAR ....	35
Ilustración 20: Logo OpenCV.....	36
Ilustración 21: Diagrama de casos de uso.....	51
Ilustración 22: Prototipo AndAR. Captura 1 .....	56
Ilustración 23: Prototipo AndAR. Captura 2 .....	56
Ilustración 24: Prototipo OpenCV. Captura 1 .....	57
Ilustración 25: Prototipo OpenCV. Captura 2.....	58
Ilustración 26: SpeechRecognizer de Google.....	59
Ilustración 27: Arquitectura de la aplicación.....	60
Ilustración 28: Diagrama de clases de la aplicación .....	61
Ilustración 29: Pantalla principal de la aplicación.....	62
Ilustración 30: Pantalla de ajustes .....	62
Ilustración 31: Pantalla de selector de color .....	63
Ilustración 32: Diagrama de navegación.....	64
Ilustración 33: Fin del soporte a ADT.....	65
Ilustración 34: Android SDK Manager.....	66
Ilustración 35: Aplicación en Google Play de OpenCV .....	66
Ilustración 36: Estructura del proyecto.....	68
Ilustración 37: Diagrama de Gantt .....	81
Ilustración 38: Pantalla de captura de video .....	87
Ilustración 39: Pantalla de ajustes .....	88
Ilustración 40: Pantalla de selector de color .....	89
Ilustración 41: Esquema de marcador de códigos QR.....	90
Ilustración 42: Ejemplo de marcador de código QR.....	91

# 1. Introducción

---

## 1.1 Motivación

La motivación para la realización de este proyecto surge tras observar ciertos aspectos de la vida cotidiana que han facilitado su realización sin un coste excesivamente alto:

-Cada vez está más extendido el uso de teléfonos inteligentes y estos, están mejorando de forma sustancial sus especificaciones, pudiendo llegar a remplazar a los ordenadores en muchas tareas.

-Está creciendo la capacidad de interconexión de los dispositivos domésticos, lo que permite obtener información de ellos y operarlos de forma remota.

Partiendo de esas dos observaciones, se ha encontrado la motivación para llevar a cabo este desarrollo, que además puede ayudar a personas con problemas de movilidad, a las cuales puede ser de apoyo en su día a día.

## 1.2 Objetivos

El objetivo principal de este proyecto es diseñar e implementar una interfaz de domótica asistencial basada en realidad aumentada, además, su implementación tratará de cumplir los siguientes objetivos secundarios:

- Integrar en un dispositivo móvil una solución que permita obtener información del estado de los aparatos conectados en la casa.
- Interactuar mediante el uso de comandos de voz para operar los aparatos
- Investigar técnicas de realidad aumentada para mostrar la información.
- Crear una solución que pueda ser usada y sirva de ayuda a personas de movilidad reducida.

## 1.3 Contexto socio-cultural y económico

En la actualidad, encontramos multitud de aparatos tecnológicos con conexión de red de serie. Sin necesidad de buscar entre tiendas especializadas en domótica, podemos encontrar desde aparatos relacionados con la informática como pueden ser impresoras, hasta dispositivos que antes no tenían una relación tan directa, que se han ido

acercando poco a poco en los últimos años, como cámaras de vigilancia, sistemas de televisión o de radio.

Si además buscamos en tiendas más especializadas, podemos llegar a encontrar soluciones que pueden abarcar desde sistemas de calefacción, electrodomésticos de uso común, simples interruptores o hasta bombillas.

Este tipo de dispositivos nos ofrecen la oportunidad de obtener información de ellos así como operar con ellos, y con el paso del tiempo su adquisición se vuelve más económica.

## 1.4 Estructura del documento

A continuación, se dará una breve explicación de cada uno de los apartados del documento, que permita al lector hacerse una idea de la información que contendrán.

**ESTADO DEL ARTE.** Contiene información acerca del contexto del proyecto, y de la investigación que se ha llevado a cabo para poder realizarlo.

**ANÁLISIS DEL SISTEMA.** Contiene un análisis detallado del sistema, identificando escenarios, usuarios, y requisitos del proyecto.

**DISEÑO DEL PROYECTO.** Hace una descripción detallada de todos los aspectos del diseño que se han decidido para llevar a cabo el proyecto, así como su arquitectura e interfaces.

**IMPLEMENTACIÓN DEL SISTEMA.** Este apartado ofrece información sobre las herramientas utilizadas, la implementación de las funcionalidades y las tecnologías utilizadas.

**PRUEBAS DEL SISTEMA.** Contiene los planes de pruebas definidos con el objetivo de asegurar la calidad del proyecto.

**GESTIÓN DEL PROYECTO.** Contiene la información acerca de la metodología utilizada en el proyecto, lo que incluye las planificaciones de las distintas fases y su presupuesto.

**CONCLUSIONES.** En esta sección se expondrán las conclusiones obtenidas tras llevar a cabo el proyecto.

**ANEXOS.** Contiene información que puede ser útil para el uso de la aplicación, como un manual de uso de la aplicación y un manual para la creación marcadores para su uso con la aplicación, e información que pueda ser útil para la comprensión del documento, incluyendo una relación de definiciones, acrónimos y referencias.

## 2. Estado del arte

---

### 2.1 Sistemas de domótica

#### 2.1.1 Introducción

Podemos empezar viendo la definición que hace la Real Academia Española para el término domótica:

“Conjunto de sistemas que automatizan las diferentes instalaciones de una vivienda.”

Además, tal y como menciona el diccionario, la palabra deriva del latín domus (casa) e informática. Sin embargo, esa definición no termina de definir completamente la domótica, al menos desde el punto de vista de la informática.

Podemos ampliar dicha definición como un conjunto de sistemas capaces de automatizar las diferentes instalaciones de la vivienda, ofreciendo servicios de gestión energética, seguridad, bienestar y comunicación.

Como se explicará a continuación, los sistemas de domótica tienen una serie de beneficios y de desventajas.

#### 2.1.2 Innovaciones

El uso de los sistemas domóticos, en pocas palabras, nos aporta confort, comunicación y seguridad, lo cual nos lleva a ganar calidad de vida.

Desde el punto de vista energético, la automatización nos permite hacer un uso eficiente de la tecnología. La gestión inteligente de la iluminación, climatización, riego, sistemas multimedia, etc, nos permite entre otras cosas reducir el consumo eléctrico, lo cual nos beneficia económicamente pero además también es bueno para el medio ambiente.

Desde el punto de vista del confort, los sistemas de domótica nos permiten realizar actuaciones, que nos repercuten en una mejora de las comodidades de una vivienda. Algunos posibles ejemplos son los siguientes:

- Apagado o encendido de las luces de la vivienda de forma inteligente, permitiendo el apagado general de todas en el caso salida de ella, adecuación de la intensidad al nivel de la luz natural ambiental o incluso llegando a la inhibición en caso de no

considerarse necesaria debido a la luz residual o por no haber nadie en la zona.

- Control remoto vía internet de los sistemas conectados en la vivienda, como puede ser la climatización, para tener una temperatura de confort en los momentos previos a nuestra llegada.
- Automatización de los sistemas bajo las necesidades específicas de los usuarios.
- Comunicación entre los distintos sistemas de la vivienda.

Además, también ofrece ventajas en el ámbito de la seguridad, ofreciendo algunas garantías a nivel personal y material:

- Alarmas de intrusión, los sensores de presencia permiten saber si hay alguien en la vivienda en todo momento, lo que nos permite saber si hay alguien cuando no debería de haber nadie.
- Cámaras IP, nos permiten ver lo que está ocurriendo en todo momento.
- Alarmas de detección, nos permiten reducir los daños derivados de incendios, fugas de gas, fugas de agua, etc.
- Teleasistencia, permite monitorizar a las personas que pueden necesitar ayuda en aquellos casos en que vivan solas.

Otra ventaja que nos aporta es la comunicación, como ya se ha comentado, los sistemas nos permiten obtener información de ellos así como transmitírsela nosotros, esto nos lleva a las siguientes ventajas en ámbito de la comunicación, algunas compartidas por los puntos anteriores:

- Ubicuidad del control, nos permite operar todos los sistemas desde un mismo control, pudiendo ser incluso desde fuera de la vivienda.
- Información de estado, nos permite evaluar el estado de los sistemas y actuar en consecuencia.
- Teleasistencias, además de en casos de emergencia también pueden ahorrar desplazamientos, la capacidad de comunicación facilita la atención a distancia.

### 2.1.3 Arquitectura

La arquitectura de los sistemas domóticos se clasifican en función de donde resida la inteligencia del sistema, hay tres variantes:

#### Arquitectura centralizada

Un controlador centralizado recibe la información de los distintos sensores, y una vez procesada la información, envía las órdenes oportunas a los dispositivos de la red.

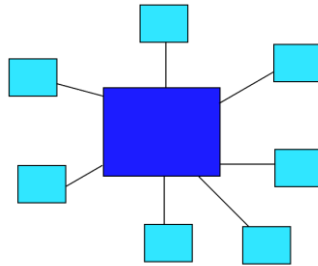


Ilustración 1: Arquitectura de sistema centralizado

#### Arquitectura descentralizada

La inteligencia del sistema reside en todos los módulos del sistema, suele ser típico de los sistemas de cableado en bus o redes inalámbricas.

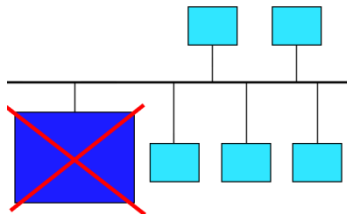


Ilustración 2: Arquitectura de sistema descentralizado

#### Arquitectura distribuida

Sistemas con arquitectura descentralizada en cuanto a que dispone de diversos dispositivos capaces de adquirir y evaluar la información de diversos sensores y retransmitírselo al resto de dispositivos.

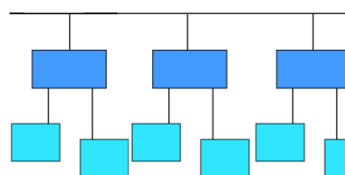


Ilustración 3: Arquitectura de sistema distribuido

## 2.1.4 Normativa y legislación

A continuación se van a mostrar las distintas normas y disposiciones legales relacionadas con la domótica, pero antes, hay que aclarar que no afectan del mismo modo. Básicamente las normas técnicas son de carácter voluntario, mientras que las disposiciones legales con frecuencia hacen referencia a las normas técnicas para cumplir los requisitos que incluyen.

### 2.1.4.1 Normas Técnicas

Existen distintos organismos autorizados para la elaboración de Normas, se distinguen entre su campo de aplicación y en su ámbito geográfico. Cabe destacar que, en España, el único organismo de normalización existente es AENOR, que cubre todos los campos de aplicación, elabora normas nacionales y contribuye y aporta su conocimiento a la elaboración de normas europeas e internacionales.

Los principales organismos son los que se muestran a continuación:

Ámbito de aplicación	General	Eléctrico	Telecomunicaciones
Internacional			
Europeo			
Nacional			

Tabla 1: Principales organismos por campo de aplicación y ámbito geográfico

Entre las principales disposiciones de carácter voluntario que afectan a las instalaciones domóticas encontramos: las familias de normas UNE-EN 50090 y la UNE-EN 50491 y una especificación técnica nacional EA0026. A continuación se muestran las principales características de cada una de ellas.

### **Normas UNE-EN 50090 para Sistemas Electrónicos de Viviendas y Edificios (HBES)**

Las normas UNE-EN 50090 normalizan las aplicaciones de control del sistema de comunicación abierto destinado a viviendas y edificios. Cubren cualquier combinación de dispositivos electrónicos conectados a través de una red de transmisión digital y tienen en cuenta los sistemas de control de automatización, tanto descentralizados como distribuidos.

Esta serie de normas se centra en la Clase I del sistema de comunicación (comunicación de datos a baja velocidad destinada al control) como por ejemplo: control del alumbrado, calefacción, gestión de energía, alarma de incendios, control de persianas, diferentes formas de control de seguridad, entre otros.

### **Normas UNE-EN 50491 para Sistemas Electrónicos de Viviendas y Edificios (HBES) y Sistemas de Automatización y Control de Edificios (BACS)**

Estas normas son independientes del protocolo de comunicación y recogen los requisitos generales de los sistemas electrónicos para viviendas y edificios (HBES) y sistemas de automatización y control de edificios. Cubren los requisitos ambientales, de compatibilidad electromagnética (CEM), seguridad eléctrica y seguridad funcional de los dispositivos y sistemas HBES y BACS.

### **Especificación EA0026 para Instalaciones de Sistemas Domóticos de Viviendas**

La especificación técnica EA0026 es un documento de rango inferior a una norma y establece los requisitos mínimos que debe cumplir el sistema domótico de Clase I, fijando las prescripciones generales de instalación y evaluación, y los diferentes niveles de domotización a nivel residencial.

Esta especificación que sirve de referencia para la Certificación de Sistemas Domóticos de Viviendas y surge con el objetivo de:

- Impulsar el desarrollo del mercado domótico.
- Aclarar la confusión existente en el mercado respecto a lo que es un sistema domótico.
- Poder comparar entre las diferentes ofertas del mercado.



### 2.1.4.2 Reglamentación de las instalaciones domóticas

Las disposiciones legales de obligado cumplimiento para las instalaciones domóticas quedan principalmente recogidas en el Reglamento Electrotécnico para Baja Tensión (REBT) y en el anexo Hogar Digital del Reglamento de Infraestructuras Comunes de Telecomunicaciones (ICT).

#### **Reglamento Electrotécnico para Baja Tensión**

El REBT aprobado por el R.D. 842/2002, del 2 de agosto, establece las condiciones técnicas y garantías que debe reunir una instalación eléctrica de baja tensión para los siguientes fines:

- Preservar la seguridad de las personas y los bienes.
- Asegurar el normal funcionamiento de dichas instalaciones y prevenir las perturbaciones en otras instalaciones y servicios.
- Contribuir a la fiabilidad técnica y a la eficiencia económica de las instalaciones.

El REBT incluye 51 instrucciones técnicas complementarias (ITC) y hace referencia a las normas UNE aplicables a los sistemas de automatización y control. La ITC-BT 51 se aplica a los sistemas de automatización y gestión técnica de la energía y seguridad para viviendas y edificios.

#### **ITC-BT 51**

La Instrucción Técnica 51 del REBT establece los requisitos mínimos de la instalación de los sistemas domóticos y comprende a las instalaciones de sistemas no independientes que realizan una función de automatización.

Sobre la ITC-BT 51 se fundamenta la Guía ITC-BT 51 que es un documento no vinculante que recoge la aplicación práctica de las previsiones del REBT y sus ITC's. En esta guía se explican y se clarifica el concepto de sistema domótico, se incluyen recomendaciones referentes a la pre-instalación domótica y se definen los niveles de domotización.

#### **Reglamento ICT: Anexo Hogar Digital**

El Anexo V del Reglamento de ICT (R.D. 345/2011) es de aplicación voluntaria y tiene como objetivo facilitar la incorporación de las funcionalidades del Hogar Digital a las viviendas, apoyándose en las soluciones que figuran en el propio Reglamento.

En este Anexo se establecen una serie de funcionalidades y niveles para clasificar un Hogar Digital como: básico, medio y superior. Su aplicación requiere de una infraestructura, de una determinada pasarela residencial y una serie de servicios.

## 2.2 Sistemas móviles

### 2.2.1 Introducción

En la actualidad, hay una gran diversidad de sistemas móviles. La sociedad se ha acostumbrado al uso de los dispositivos móviles, de diversos tipos, y los utilizan en el día a día, dándose el caso de algunas personas, que llegar a desarrollar una dependencia hacia ellos.

El sistema operativo de un dispositivo móvil, es el programa base que hace funcionar a dicho dispositivo. Si estuviéramos hablando de ordenadores de sobremesa, sus homólogos serían Windows, Mac, Linux, etc. Y al igual que con dichos SSOO, los SSOO de los dispositivos móviles están en constante evolución, y compiten entre ellos para acaparar el mayor número de usuarios.

La tendencia de los SSOO actuales, es la de permitir hacer desarrollos y mejoras, creando de ese modo un negocio de venta de aplicaciones, que aunque no logran gestionar íntegramente, si son capaces de sacar beneficio de ello. Los mayores ejemplos son la App Store de iOS y Google Play de Android, que intentan canalizar y unificar en sus respectivas tiendas virtuales, todas las aplicaciones que los usuarios puedan necesitar en sus móviles inteligentes. Sin embargo, existen alternativas, en algunos casos legítimas, como el caso de la Amazon Appstore, que pone a disposición de los usuarios aplicaciones que muchas veces también están publicadas en Google Play.

No obstante, también existen repositorios de dudosa legalidad, que basan su creación en el pretexto de crear copias para uso privado, sin embargo cualquiera puede obtener dichas aplicaciones, que en algunos casos pueden no ser las originales y tienen la intención de infectar el dispositivo con algún virus.

### 2.2.2 Dispositivos móviles

Cada vez son más comunes los dispositivos móviles, y aunque lo normal es que casi todos los móviles modernos esté conectados a internet, no es un requisito indispensable para su funcionamiento, aunque si puede limitar bastantes de sus funcionalidades.

En este caso, entendemos por dispositivo móvil, todo aparato de reducidas dimensiones, con capacidad de procesamiento y almacenamiento, con posibilidad de conexión a internet y diseñados con el objetivo de facilitar su transporte.

Aunque la teoría es que los dispositivos móviles son versiones reducidas de los ordenadores domésticos, en el mercado nos podemos encontrar dispositivos móviles, que superan a algunos ordenadores en precio, así como en prestaciones y capacidad de computación. De hecho,

las últimas tendencias en los móviles, llevan a aumentar el número de núcleos de sus procesadores con el objetivo de aumentar sus prestaciones sin hacer peligrar la autonomía de los dispositivos.

Tener más núcleos no significa tener mejor procesador, pero en la actualidad, los buques insignia de los principales fabricantes no bajan de los 8 núcleos. Concretamente, el Exynos™ 7 Octa 7580 de Samsung y el Snapdragon™ 810 de Qualcomm contienen en su interior 8 núcleos, y Mediatek con su Helio™ X20 llega a los 10 núcleos. Sin embargo un móvil que tenga un procesador con más núcleos no implica que funcione mejor, y la culpa de ello en parte la tiene el sistema operativo.

### 2.2.3 Aplicaciones móviles

La aparición de los teléfonos inteligentes y de los SSOO que dan la posibilidad de desarrollar aplicaciones, ha provocado un cambio en el uso de los ordenadores. Cada vez más personas han dejado de usar el ordenador para hacer tareas que hasta su aparición era el único modo de llevarlas a cabo, como puede ser leer y mandar correos electrónicos, ver fotos/videos, o utilizar las redes sociales. La cantidad de aplicaciones crece continuamente y aunque nos facilitan la vida, a veces pueden ser un riesgo.

Estas aplicaciones, nos han llevado a las personas a depositar en los dispositivos móviles gran cantidad de datos personales, que no siempre se protegen adecuadamente, tanto por parte de los desarrolladores como por parte de los usuarios, y pueden suponer un problema en caso de pérdida de los dispositivos.

Ya no hablamos de los recuerdos familiares, la música que escuchemos o la agenda de teléfonos que almacenemos en los móviles, que también son importantes, las facilidades que nos aportan los móviles, nos han llevado a utilizarlo para cosas aún más importantes, como puede ser la realización de operaciones bancarias. Por eso es importante, a la hora de instalar y utilizar las aplicaciones en los móviles, tener las debidas precauciones y no confiar en software obtenido de fuentes no fiables.

### 2.2.4 Sistemas operativos

Los sistemas operativos, son un programa o conjunto de ellos, instalados en un sistema informático, que se encargan de gestionar los recursos hardware y de proveer servicios a los programas de aplicación. El SO se ejecuta en modo privilegiado, aunque puede que parte de él se ejecute en espacio de usuario.

Lo más esencial de un sistema operativo es el núcleo o también llamado kernel. En este caso, nos interesa los que gobiernan los

dispositivos móviles, que están optimizados para reducir el consumo de recursos y así aumentar su eficiencia energética.

Existe gran variedad de S.O.O. móviles, cada uno tiene sus ventajas e inconvenientes, pero en la actualidad hay 4 grandes competidores, que se reparten el 99% de los dispositivos móviles. La dominación del mercado es muy volátil y en menos de una década S.O.O dominantes puedes desaparecer dejando hueco a viejas glorias o a nuevos competidores.

A continuación, se expondrán los cuatro sistemas operativos móviles que compiten actualmente por innovar, captar más usuarios y robárselos a sus oponentes:

- **Android.** Este sistema operativo ha sido el más dominante en los últimos años, en parte, por ser de código libre y abierto. Existe un gran mercado de aplicaciones, que comprende tanto contenido gratuito, como de pago. No siempre fue así, pero en la actualidad es propiedad de Google y gracias a sus innovaciones y su respaldo, ha llevado a Android a estar instalado en 4 de cada 5 dispositivos móviles.
- **iOS.** Es el sistema operativo móvil propiedad de la multinacional Apple Inc. Se desarrolló inicialmente para el iPhone, con el nombre de iPhone OS, para después usarse también en los iPod touch y en el iPad, y así cambiar su nombre a iOS. Es el segundo SO móvil más usado y al contrario que Android, está implantado únicamente en los dispositivos fabricados por Apple Inc.
- **Windows Phone.** Es el sistema operativo móvil desarrollado por Microsoft. Han sacado versiones paralelas a los sistemas operativos de escritorio, aunque se espera que con la llegada de Windows 10, se unifiquen en una sola versión. Al contrario que su predecesor, no está enfocado al mercado empresarial, si no al mercado de consumo, y gracias a la integración de sus servicios propios como OneDrive, Skype y Xbox Live, poco a poco está logrando aumentar el número de usuarios que hacen uso de él.
- **BlackBerry 10.** Este sistema operativo es el más veterano de los 4 competidores. Su primera versión fue en el año 1999, hace ya 16 años y aunque ha perdido gran parte de sus usuarios, lucha por recuperar su lugar en el podio de los S.O.O

móviles más usados. Las versiones anteriores llevan el nombre BlackBerry OS, dando el salto de la versión 7 a la 10. Esta versión, trajo entre otras novedades, la compatibilidad con aplicaciones Android, pudiendo hacer uso de aplicaciones desarrolladas para Android.

Las siguientes tablas muestran los datos obtenidos por la International Data Corporation (IDC) acerca de la implantación en los teléfonos móviles:

Period	Android	iOS	Windows Phone	BlackBerry OS	Others
Q3 2014	84.4%	11.7%	2.9%	0.5%	0.6%
Q3 2013	81.2%	12.8%	3.6%	1.7%	0.6%
Q3 2012	74.9%	14.4%	2.0%	4.1%	4.5%
Q3 2011	57.4%	13.8%	1.2%	9.6%	18.0%

Tabla 2: Cuotas de mercado de SSOO móviles

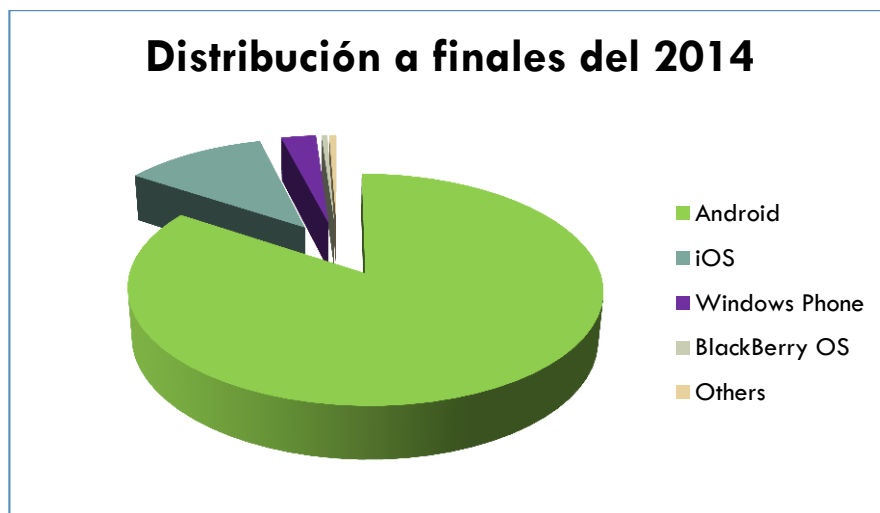


Ilustración 4: Gráfica de cuota de mercado

## 2.3 Android

### 2.3.1 Introducción

Se trata de un sistema operativo basado en Linux, que en sus comienzos se desarrolló para ser utilizado en teléfonos móviles y en la actualidad ha pasado a formar parte también de las tabletas, relojes inteligentes, televisores e incluso automóviles.

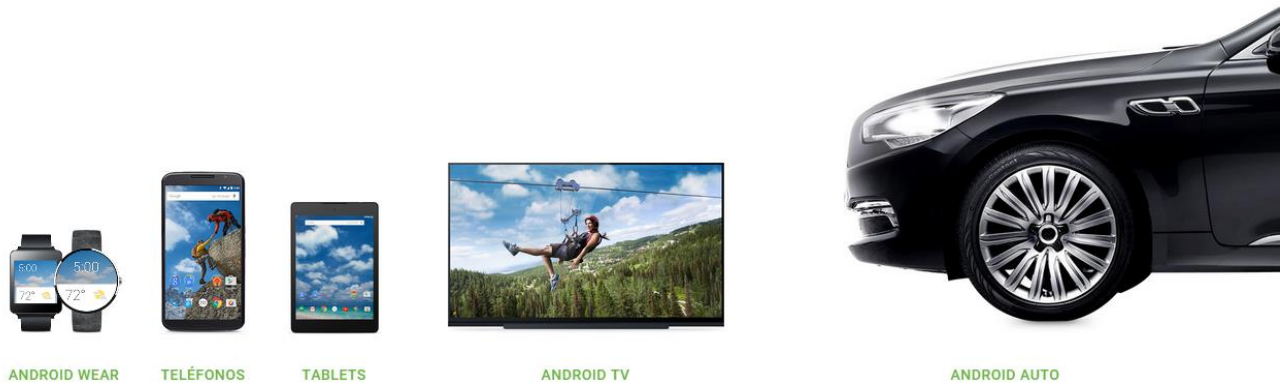


Ilustración 5: Implantación de Android en otros dispositivos

Comenzó como un proyecto de Android Inc., empresa apoyada económicamente por Google, pero en el año 2005 pasó a formar parte su propiedad, cuando decidió comprarla.

Un par de años más tarde, en el 2007, Google junto con otras 84 compañías, entre las que se encuentran fabricantes de teléfonos móviles como HTC o Samsung, o de procesadores como Qualcomm o Texas instrument, forman la Open Handset Alliance. Su objetivo es formar una alianza comercial para desarrollar estándares abiertos para dispositivos móviles.

Gracias a ese acuerdo, lograron impulsar la implantación de Android, que ayudado también por su código libre y abierto, consiguieron un gran crecimiento en poco tiempo, ya que muchos fabricantes de teléfonos lo adoptaron como sistema operativo para sus dispositivos.



Ilustración 6: Evolución de los dispositivos con SO Android

Google puso a disposición de desarrolladores y usuarios, una plataforma donde unos publicaban sus aplicaciones y otros instalaban o compraban sus desarrollos. Esta plataforma recibió en sus inicios el nombre de Android Market, pero posteriormente pasó a llamarse Google Play. En Julio de 2015 la cifra de aplicaciones instaladas superó la cantidad de 1.6 millones.



Ilustración 7: Evolución de Android Market

### 2.3.2 Arquitectura de Android

En los comienzos de Android, estaba construido con la máquina virtual Dalvik, sin embargo en las últimas versiones, el runtime de Android ha pasado a ser ART.

Con la llegada de ART, se pretende mejorar el rendimiento de las aplicaciones, una de sus mejoras es que en lugar de compilar las aplicaciones cada vez que se ejecutan, como hacía Dalvik, se dejan compiladas en el momento de la instalación. Esto conlleva que el tiempo de ejecución sea menor pero que aumente el espacio que ocupan las aplicaciones. Además, también incluyó mejoras en el recolector de



memoria, en las herramientas para depurar código y mejoras en el diagnóstico de excepciones.

Además, Android está construido por diversas capas que persiguen facilitar el desarrollo de aplicaciones y proteger la seguridad de los dispositivos, a continuación se enumerarán las diferentes capas:

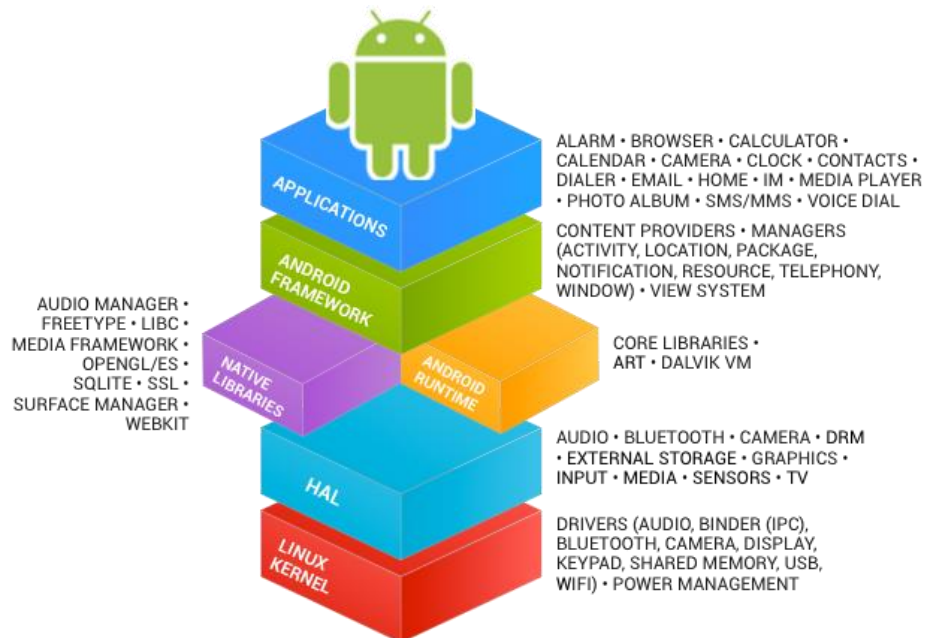


Ilustración 8: Capas de la arquitectura Android

### Framework de aplicaciones

Es el framework que utilizan los desarrolladores para obtener información de las aplicaciones base del sistema. Está diseñado para simplificar su reutilización, cualquier aplicación puede hacer uso de su API y además su API es retrocompatible con versiones antiguas.

### Binder IPC

Es la herramienta que Android pone a disposición de los desarrolladores para comunicarse con los servicios de Android y acceder a información que no sea de su aplicación.

### Servicios de sistema

Funcionalidad expuesta por el framework de aplicaciones y que se comunica con el hardware del sistema. Están contruidos de forma modular y están clasificados en dos grupos, los de Sistema y los de Servicios de comunicación. Algunos ejemplos son el Servicio de búsqueda, el Gestor de notificaciones o Gestor de ventanas.

## Capa de abstracción de hardware (HAL)

Es una interfaz estándar que permite al sistema Android comunicarse con el hardware. Contiene las interfaces comunicación con los drivers de los componentes del dispositivo.

## Nucleo del sistema:

Android utiliza un kernel de Linux con modificaciones especiales como los 'wake locks', el driver para poder realizar la comunicación entre aplicaciones y otras implementaciones necesarias para su uso en dispositivos móviles.

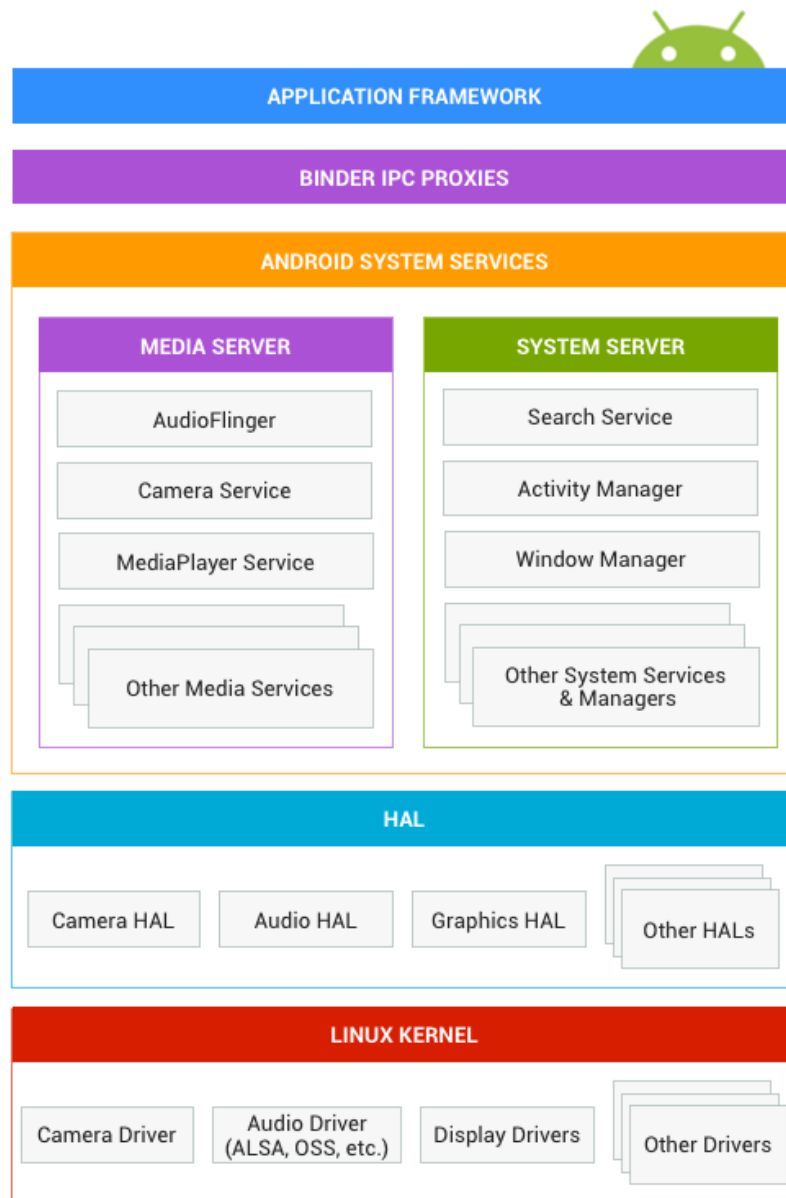


Ilustración 9: Estructura de arquitectura Android

### 2.3.3 Versiones de Android

Desde sus comienzos, Android no ha dejado de innovar, sacar novedades y nuevas versiones de su sistema operativo. Con cada actualización mayor recibe un nombre nuevo, cada vez comenzando con la siguiente letra del abecedario, y un nuevo número de API, si por el contrario solo es una actualización menor únicamente aumenta su número de API, manteniendo el nombre de versión.

Los nombres que ha recibido desde su versión 1.0 han sido postres o dulces, y se espera que los que le sigan también lo sean. A continuación se muestra la lista completa de versiones y su nombre clave:

Letra	Nombre	Traducción
<a href="#"><u>A</u></a>	Apple Pie (v1.0)	Tarta de manzana
<a href="#"><u>B</u></a>	Banana Bread (v1.1)	Pan de plátano
<a href="#"><u>C</u></a>	Cupcake (v1.5)	Panque
<a href="#"><u>D</u></a>	Donut (v1.6)	Rosquilla
<a href="#"><u>E</u></a>	Éclair (v2.0/v2.1)	Pepito
<a href="#"><u>F</u></a>	Froyo (v2.2)	Yogurt helado
<a href="#"><u>G</u></a>	Gingerbread (v2.3)	Pan de jengibre
<a href="#"><u>H</u></a>	Honeycomb (v3.0/v3.1/v3.2)	Panal de miel
<a href="#"><u>I</u></a>	Ice Cream Sandwich (v4.0)	Sandwich de helado
<a href="#"><u>J</u></a>	Jelly Bean (v4.1/v4.2/v4.3)	Gominola
<a href="#"><u>K</u></a>	KitKat (v4.4)	Kit Kat
<a href="#"><u>L</u></a>	Lollipop (v5.0/v5.1)	Piruleta

Tabla 3: Relación de nombres de versiones Android

Google pone a disposición de todo el mundo, los datos sobre la distribución de sus últimas versiones. Los datos los obtiene de los accesos a Google Play, que solo es compatible con las versiones 2.2 en adelante y se actualizan regularmente. A continuación, se pueden ver los datos obtenidos de la semana del 3 de Agosto de 2015:

Version	Codename	API	Distribution
<a href="#"><u>2.2</u></a>	Froyo	8	0.3 %

<a href="#">2.3.3 - 2.3.7</a>	Gingerbread	10	4.6 %
<a href="#">4.0.3 - 4.0.4</a>	Ice Cream Sandwich	15	4.1 %
<a href="#">4.1.x</a>	Jelly Bean	16	13 %
<a href="#">4.2.x</a>		17	15.9 %
<a href="#">4.3</a>		18	4.7 %
<a href="#">4.4</a>	KitKat	19	39.3 %
<a href="#">5.0</a>	Lollipop	21	15.5 %
<a href="#">5.1</a>		22	2.6 %

Tabla 4: Distribución de versiones Android en el año 2015

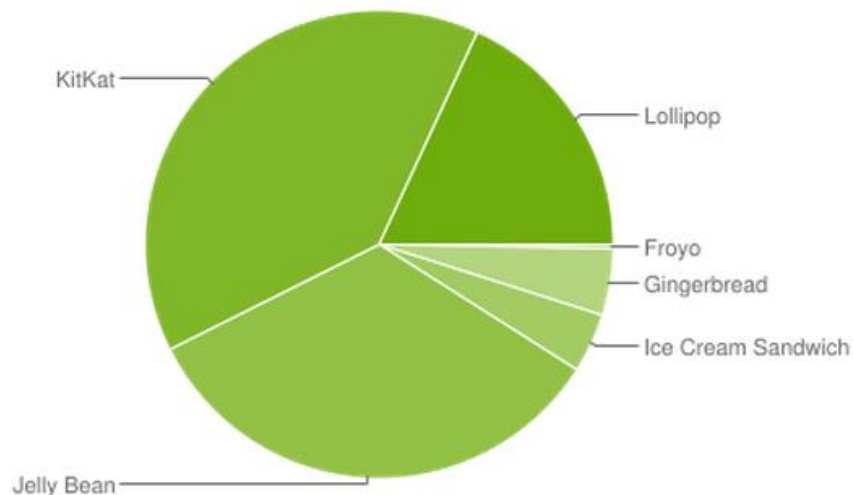


Ilustración 10: Distribución de versiones Android en el año 2015

### 2.3.4 Forks de Android

Con el beneplácito de Android, otras compañías o simples desarrolladores han hecho uso del código fuente que Google pone a disposición, para sacar versiones de Android modificadas. Google permite estas prácticas siempre que no incluyan el uso de las partes que no son de código libre, como pueden ser las aplicaciones de Gmail, Google Maps y demás código privativo.

Existen versiones que han llegado a lograr gran fama, llegando incluso a venir preinstaladas en móviles, como puede ser el caso de

Cyanogen OS, que venía instalada en su versión 12 en los móviles OnePlus One de la empresa china OnePlus.

Existen gran cantidad de versiones, casi todas ellas intentan mejorar el rendimiento de Android permitiendo llevar versiones más modernas a teléfonos que han quedado sin soporte por su antigüedad.

Otras compañías sin embargo, lo hacen para adaptarlo a sus necesidades y a su modelo de negocio, como es el caso de Amazon, que sacó su versión de Android, Fire OS, y con ella su propio 'Market' de aplicaciones.

Por su parte, Google también sacó una versión modificada, buscando llevar móviles de bajo presupuesto con versiones adaptados a sus capacidades, este proyecto se llamó Android One.

Hay muchas versiones, algunos fabricantes de móviles han creado la suya propia, y estas versiones modificadas de Android se están extendiendo, y aunque antes estaban enfocadas a usuarios expertos, ahora viene de fábrica en algunos teléfonos y son más fáciles de instalar. Según un estudio de la empresa ABIresearch, en 2014 llegaron a suponer el 20% del total de dispositivos Android.

## 2.4 Códigos QR

### 2.4.1 Información general

Los códigos QR fueron creados por la subsidiaria DENSO WAVE, propiedad de Toyota, en el año 1994. Esta empresa japonesa, se dedica a crear dispositivos de identificación automática.

Estos códigos bidimensionales o matriciales permiten la codificación de información en un espacio reducido, es capaz de manejar caracteres numéricos y alfanuméricos, símbolos, datos binarios y códigos de control, llegando a codificar hasta un límite de 7089 caracteres en un único código QR de la versión de más tamaño (v. 40).

A mayor tamaño del código QR, mayor capacidad de codificación:

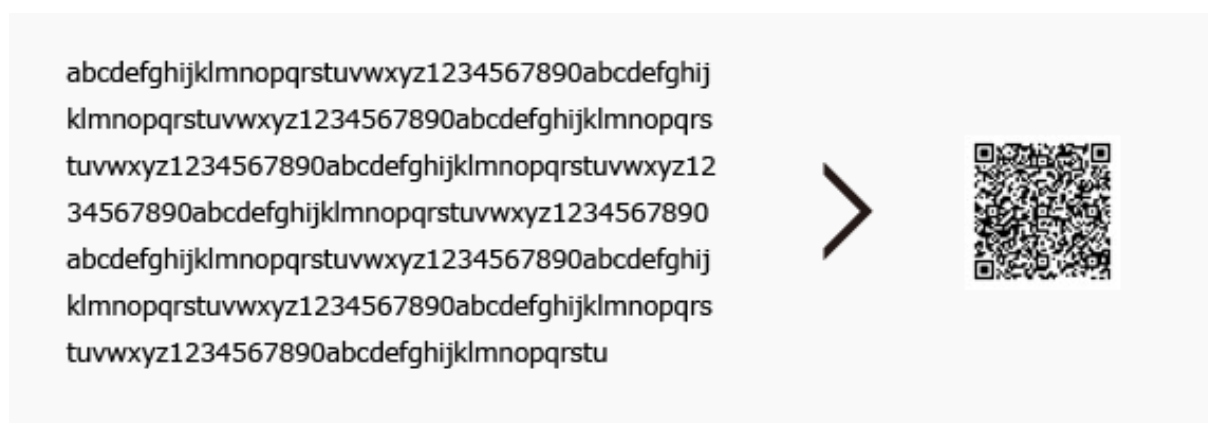


Ilustración 11: Código QR con capacidad de 300 caracteres alfanuméricos

### 2.4.2 Ventajas

Tolerancia a errores.

El diseño del código presenta cierta tolerancia a errores, pudiendo ser leído cuando el código no es legible al 100%. Existen 4 niveles de corrección, los cuales hacen que el código deba ser de mayor tamaño a media que se aumenta el nivel de corrección:

<b>Nivel L (Low)</b>	Aprox. 7%
<b>Nivel M (Medium)</b>	Aprox. 15%
<b>Nivel Q (Quartile)</b>	Aprox. 25%
<b>Nivel H (High)</b>	Aprox. 30%

Tabla 5: Tabla de tolerancia a errores QR

Lectura en cualquier dirección (360%)

La estructura del código permite detectar la posición del código para realizar una correcta decodificación. Como puede observarse en la imagen, se identifican 3 cuadrados de mayor tamaño que son los que hacen posible la orientación del código:

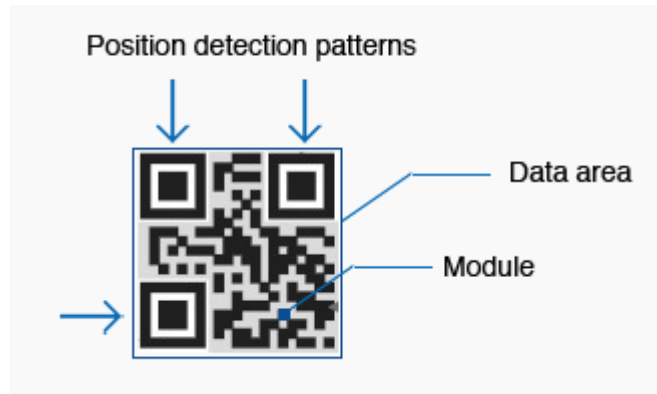


Ilustración 12: Sistema de orientación del código QR

Tamaño en función de las necesidades

Dependiendo de la información que sea necesario codificar, se puede emplear el diseño de las primeras versiones, el cual es de menor tamaño, o de las últimas:

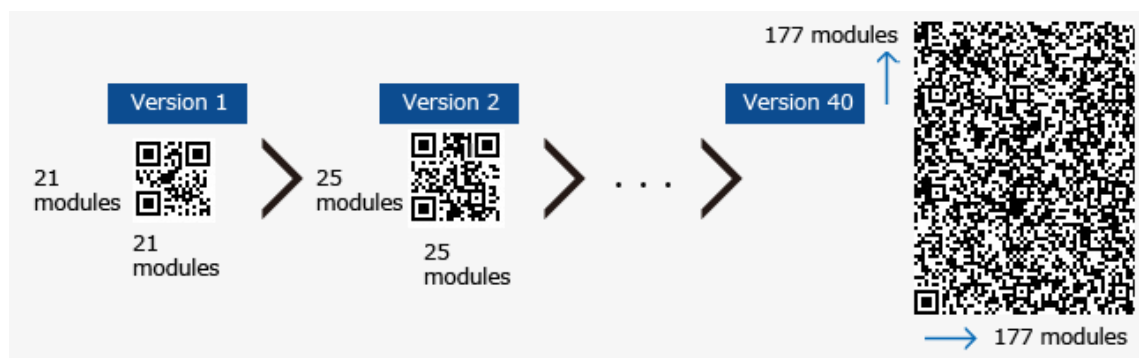


Ilustración 13: Comparativa de versiones y tamaño

## 2.4.3 Ejemplos de uso

### 2.4.3.1 Ampliación de información

Cada vez son más comunes ver estos códigos acompañando a carteles de información o de publicidad, lo cual nos amplía la información. Por ejemplo, en las paradas de autobús, nos facilitan acceso a horarios o mapas de dichos autobuses.

### 2.4.3.2 Programa piloto ‘Código QR’ para enfermos de Alzheimer

En Septiembre de 2014, el ayuntamiento de Madrid anunciaba una iniciativa para identificar y ayudar a enfermos de Alzheimer mediante el uso de estos códigos.

La iniciativa consistía en la impresión de los códigos QR en colgantes, pegatinas, chapas o pulseras, los cuales contendrán información importante sobre el enfermo. Mediante los datos almacenados, se les podrá ayudar en caso de extraviarse o desorientarse, mejorando su calidad de vida.



Ilustración 14: Programa piloto para enfermos de Alzheimer

### 2.4.4 Librería ZXing

La librería ZXing es una herramienta de código abierto para la detección de códigos de barras multiformato 1D/2D, hecha en Java y portada a otros lenguajes.

1D product	1D industrial	2D
UPC-A	Code 39	<b>QR Code</b>
UPC-E	Code 93	Data Matrix
EAN-8	Code 128	Aztec (beta)
EAN-13	Codabar	PDF 417 (beta)
	ITF	
	RSS-14	
	RSS-Expanded	

Tabla 6: Formatos compatibles con la librería ZXing

Según indican en su página web, el término ZXing deriva de “*cebra crossing*” de ahí su logo:





Ilustración 15: Logo de ZXing

La última actualización de la librería es la 3.2.0 y fue publicada el 15 de Febrero de 2015, se puede obtener de la página oficial y se distribuye bajo la licencia Apache v2.0.

## 2.5 Realidad Aumentada

El término ‘realidad aumentada’ se le atribuye a Tom Caudell, antiguo investigador de Boeing, en el año 1990, al describir una pantalla digital utilizada por los electricistas de las aeronaves para aunar gráficos virtuales y la realidad física.

La definición más extendida en el mundo de la informática es esencialmente lo que describió, pero más detallada: *“Es la interacción de los gráficos superpuestos, audio y otras mejoras de los sentidos añadido a un ambiente del mundo real que se muestran en tiempo real”*.

Esto significa que no hay que confundir la realidad aumentada con la realidad virtual, en la cual, toda la realidad es sustituida, mientras que en el caso de la realidad aumentada simplemente se añade información a la existente.

### 2.5.1 ARToolKit

Es una librería de software multiplataforma para el desarrollo de aplicaciones de realidad aumentada. Puede usarse para uso no comercial bajo licencia GNU GPL, la licencia para uso comercial es administrada por ARToolworks, Inc.

Uno de los problemas que se encuentran a la hora de superponer información, es localizarla en el espacio real. Para solucionar ese problema, ARToolkit utiliza algoritmos que realizan un seguimiento de video capaz de calcular la posición de la cámara y la orientación relativa de los objetos, y para ello hace uso de marcadores.

Los marcadores son dibujos que se caracterizan por ser una forma libre dentro de un cuadrado negro sobre un fondo blanco, de este modo con el contraste es más fácil identificarlos.



Ilustración 16: Marcador de ARToolkit

Aunque el software es capaz de identificar cualquier marcador siempre que tenga un contorno cuadrado, el algoritmo lo que hace es descomponer el dibujo en una matriz de cuadrados, con lo cual será más preciso y tendrá menos errores, cuando los marcadores siguen dicho patrón.



Ilustración 17: Marcador original, con matriz de 64x64 y de 16x16

Como es lógico, a menor complejidad del marcador, menos errores habrá en su seguimiento.

### 2.5.1.1 Proyecto AndAR

AndAR es un proyecto que permite usar la Realidad Aumentada en la plataforma Android. Todo el proyecto se distribuye bajo la Licencia Pública General de GNU. Esto significa que se puede utilizar libremente en cualquiera de sus proyectos, siempre y cuando utilice la misma licencia, la GPL. Como se trata de un proyecto de código abierto las contribuciones son bienvenidas.

El proyecto se basa en el uso de las librerías ARToolKit bajo el sistema operativo Android, la cual ofrece una aplicación básica, capaz de detectar códigos de realidad aumentada y superponer información:

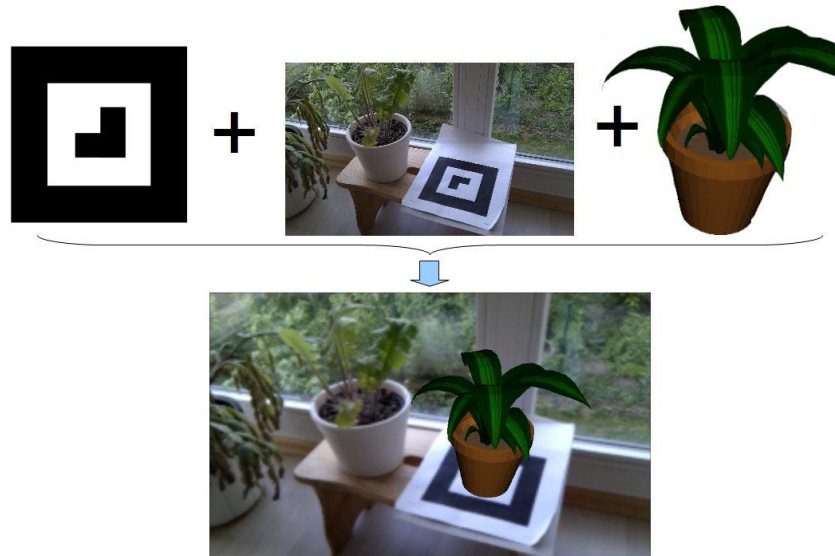


Ilustración 18: Ejemplo de realidad aumentada del proyecto AndAR



Ilustración 19: Ejemplo de realidad aumentada del proyecto AndAR

El código disponible para descargar y modificar viene con 3 modelos simples, que van asociados a 3 códigos distintos, sin embargo es posible añadirle modelos nuevos y a los que asociarles nuevos códigos.

Aunque la aplicación funciona, no es estable al 100% y la última actualización es del año 2010, por lo que no se espera que el desarrollador que inició el proyecto, Tobias Domhan, publique nuevas versiones.

## 2.5.2 OpenCV

OpenCV es una librería de código abierto para tratamiento de imágenes y aprendizaje automático. Fue construida para proporcionar herramientas a aplicaciones de visión computarizada y para acelerar el uso de la percepción de las máquinas en productos comerciales.

Cuenta con licencia BSD, lo cual permite su uso incluso para productos comerciales.

La librería cuenta con más de 2.500 algoritmos optimizados, que permiten detectar y reconocer caras, identificar objetos, clasificar acciones humanas en video, extraer modelos 3D de objetos, reconocer el paisaje y establecer marcadores para hacer uso de la realidad aumentada.

Aunque está escrita de forma nativa en C++, cuenta con interfaces de Python, Java y MATLAB, además es compatible con Windows, Linux, Android y Mac OS.

OpenCv se centra principalmente en las aplicaciones de visión en tiempo real y se aprovecha de instrucciones MMX y SSE cuando están disponibles, y actualmente se está desarrollando para hacer uso de CUDA y OpenCL.

El proyecto cuenta con mucha documentación y gran soporte, la última actualización estable, la 3.0, fue lanzada el 4 de Junio del 2015 y cuenta con versiones para Windows, Linux/Mac, Android e IOS.



Ilustración 20: Logo OpenCV

## 3. Análisis del sistema

---

En este apartado se detallarán los requisitos software definidos por el cliente a la hora de desarrollar la aplicación y sobre los que se definirá el diseño de la aplicación.

### 3.1 Identificación de escenarios y usuarios

La aplicación que se va a diseñar y desarrollar contará con una funcionalidad que permitirá a los usuarios visualizar diferentes comandos, que estarán disponibles en función de la configuración que tengan hecha en su hogar domótico.

No hay un escenario único sobre el que hacer el diseño y estará abierto a la incorporación de nuevos dispositivos.

Inicialmente solo habrá un tipo de usuario que podrá usar e interactuar con todos los comandos que haya disponibles a su alcance, aunque no queda descartada la inclusión de roles o limitaciones para futuras mejoras de la aplicación.

### 3.2 Requisitos de Usuario

En este apartado se van a detallar los requisitos definidos por el usuario, tienen como objetivo delimitar la funcionalidad que se necesita.

Se clasifican en dos categorías, requisitos de capacidad, los cuales definen las funcionalidades del sistema, y requisitos de restricción, que definirá limitaciones de la aplicación.

#### 3.2.1 Prototipo de requisito de usuario

A continuación se especificará el formato que tendrán los requisitos de usuario, detallando los campos que tendrán.

**Identificador:** incluye el tipo de requisito y un número único que identifica a cada uno de los requisitos, los números son consecutivos comenzando por el 1 y la numeración es independiente para cada tipo de requisito. Nomenclatura:

- **De Capacidad:** RUC - [ i ], siendo [ i ] el número de requisito.
- **De Restricción:** RUR - [ j ], siendo [ j ] el número de requisito.

**Nombre:** Definición más amigable del requisito que lo describa.

**Prioridad:** Define cuál de los siguientes estados tiene el requisito, en función de su importancia:

- **Requerido**
- **Deseable**
- **Opcional**

**Descripción:** Definición clara y concisa del objetivo del requisito.

**Cambios:** Registro de cambios que haya sufrido el requisito durante el desarrollo.

Requisito [Identificador]	Nombre	[Nombre]
Prioridad	[ ] Requerido	[ ] Deseable [ ] Opcional
Descripción	[Descripción]	
Cambios	[Cambios]	

Tabla 7: Prototipo de requisito

### 3.2.2 Requisitos de capacidad

A continuación se detallan los requisitos de usuario de capacidad:

Requisito	RUC - 1	Nombre	Mostrar información sobre los dispositivos
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Descripción	Mostrar información sobre los dispositivos del hogar.		
Cambios	Versión inicial: 01/04/2015		

Tabla 8: Requisito de Usuario de Capacidad 1. Mostrar información sobre los dispositivos

Requisito	RUC - 2	Nombre	Detectar comandos de voz
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional

<b>Descripción</b>	La aplicación reconocerá comandos de voz.
<b>Cambios</b>	Versión inicial: 01/04/2015

Tabla 9: Requisito de Usuario de Capacidad 2 - Detectar comandos de voz

Requisito	RUC - 3	Nombre	Leer etiquetas
<b>Prioridad</b>	<input checked="" type="checkbox"/> Requerido	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
<b>Descripción</b>	La aplicación detectará y leerá etiquetas impresas.		
<b>Cambios</b>	Versión inicial: 01/04/2015		

Tabla 10: Requisito de Usuario de Capacidad 3 - Leer etiquetas

Requisito	RUC - 4	Nombre	Mostrar información útil
<b>Prioridad</b>	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
<b>Descripción</b>	La aplicación dará ayudas acerca de las operaciones disponibles.		
<b>Cambios</b>	Versión inicial: 01/04/2015		

Tabla 11: Requisito de Usuario de Capacidad 4 - Mostrar información útil

Requisito	RUC - 5	Nombre	Obtener información del servidor
<b>Prioridad</b>	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
<b>Descripción</b>	La aplicación obtendrá las operaciones disponibles consultando al servidor.		
<b>Cambios</b>	Versión inicial: 01/04/2015		

Tabla 12: Requisito de Usuario de Capacidad 5 - Obtener información del servidor

Requisito	RUC - 6	Nombre	Menú de ajustes
<b>Prioridad</b>	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
<b>Descripción</b>	La aplicación tendrá un menú de ajustes para personalizar la aplicación.		
<b>Cambios</b>	Versión inicial: 01/04/2015		

Tabla 13: Requisito de Usuario de Capacidad 6- Menú de ajustes

Requisito	RUC - 7	Nombre	Dirección del servidor editable
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Descripción	La aplicación podrá cambiar de servidor fácilmente.		
Cambios	Versión inicial: 01/04/2015		

Tabla 14: Requisito de Usuario de Capacidad 7- Dirección del servidor editable

Requisito	RUC - 8	Nombre	Formato de información editable
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Descripción	La aplicación permitirá ajustar el formato de las letras. Permitirá ajustar: <ul style="list-style-type: none"> <li>- Tamaño</li> <li>- Grosor</li> <li>- Color</li> <li>- Color de sombra</li> </ul>		
Cambios	Versión inicial: 01/04/2015		

Tabla 15: Requisito de Usuario de Capacidad 7- Formato de información editable

Requisito	RUC - 9	Nombre	Zoom de video
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Descripción	La aplicación permitirá aumentar y disminuir el zoom del video.		
Cambios	Versión inicial: 01/04/2015		

Tabla 16: Requisito de Usuario de Capacidad 9- Zoom de video



### 3.2.3 Requisitos de restricción

A continuación se detallan los requisitos de usuario de restricción:

Requisito	RUR - 1	Nombre	Utilización con internet
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Descripción	La aplicación no podrá funcionar sin conexión a internet		
Cambios	Versión inicial: 01/04/2015		

Tabla 17: Requisito de Usuario de Restricción 1 - Funcionamiento con internet

Requisito	RUR - 2	Nombre	Funcionamiento en Android
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Descripción	La aplicación solo funcionará en dispositivos con Android.		
Cambios	Versión inicial: 01/04/2015		

Tabla 18: Requisito de Usuario de Restricción 2 - Funcionamiento en Android

Requisito	RUR - 3	Nombre	Comunicación con servidor local
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Descripción	La aplicación necesitará un servidor que se encuentre en la misma red local.		
Cambios	Versión inicial: 01/04/2015		

Tabla 19: Requisito de Usuario de Restricción 3 - Comunicación con servidor local

Requisito	RUR - 4	Nombre	Uso de cámara
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Descripción	La aplicación funcionará en un dispositivo con cámara de video.		
Cambios	Versión inicial: 01/04/2015		

Tabla 20: Requisito de Usuario de Restricción 4 – Uso de cámara

Requisito	RUR - 5	Nombre	Utilización de microfono
Prioridad	<input checked="" type="checkbox"/> Requerido	<input type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Descripción	La aplicación funcionará en un dispositivo con micrófono		
Cambios	Versión inicial: 01/04/2015		

Tabla 21: Requisito de Usuario de Restricción 5- Utilización de micrófono

Requisito	RUR - 6	Nombre	Funcionamiento con librerías estándar
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Descripción	La aplicación hará uso de librerías lo más estándar posible.		
Cambios	Versión inicial: 01/04/2015		

Tabla 22: Requisito de Usuario de Restricción 6 - Uso de librerías estándar

Requisito	RUR - 7	Nombre	Uso sin contacto
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Descripción	Para la realización de las operaciones no será necesario interactuar con la pantalla		
Cambios	Versión inicial: 01/04/2015		

Tabla 23: Requisito de Usuario de Restricción 7 - Uso sin contacto

Requisito	RUR - 8	Nombre	Sin datos de operaciones
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Descripción	La aplicación no almacenará la información de las operaciones y dispositivos.		
Cambios	Versión inicial: 01/04/2015		

Tabla 24: Requisito de Usuario de Restricción 8 - Uso sin datos

### 3.3 Requisitos de software

En este apartado se van a detallar el conjunto de requisitos que se van a tener en cuenta durante el desarrollo de la aplicación. Dichos requisitos se van a clasificar en requisitos funcionales y requisitos no funcionales.

La definición de los requisitos permitirá fijar un alcance del proyecto, y en el caso de que el cliente no esté de acuerdo con algún punto del diseño, se podrá saber si entra dentro de su presupuesto o si por el contrario queda fuera del alcance.

La especificación de los requisitos se clasificará en los siguientes subconjuntos:

- **Funcionales:** Son aquellos que definen la funcionalidad de la aplicación y determinan aquellas acciones que puedan realizar los usuarios que hagan uso de ella.
- **No funcionales:** Son aquellos que definen las restricciones o condiciones en aspectos de calidad, estándares, costes, estabilidad y portabilidad.

#### 3.3.1 Prototipo de requisito

A continuación se especificará el formato que tendrán los requisitos, detallando los campos que tendrán.

**Identificador:** incluye el tipo de requisito y un número único que identifica a cada uno de los requisitos, los números son consecutivos comenzando por el 1 y la numeración es independiente para cada tipo de requisito. Nomenclatura:

- **Funcionales:** F - [ i ], siendo [ i ] el número de requisito.
- **No Funcionales:** NF - [ j ], siendo [ j ] el número de requisito.

**Fuente:** define el requisito de usuario que ha generado el requisito de software

El resto de información es común a los requisitos de usuario.

### 3.3.2 Requisitos funcionales

A continuación se detallarán los requisitos funcionales definidos por el cliente:

Requisito	F - 1	Nombre	Mostrar información sobre la captura
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Fuente	RUC -1		
Descripción	La aplicación mostrará la información que necesite el usuario superpuesta a la captura del video.		
Cambios	Versión inicial: 01/04/2015		

Tabla 25: Requisito funcional 1 - Mostrar información sobre la captura

Requisito	F - 2	Nombre	Interactuar con comandos de voz
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Fuente	RUC - 2		
Descripción	La aplicación reconocerá comandos de voz. La aplicación será capaz de identificar la operación a realizar tras reconocer el comando de voz del usuario.		
Cambios	Versión inicial: 01/04/2015		

Tabla 26: Requisito funcional 2 - Interactuar con comandos de voz

Requisito	F - 3	Nombre	Reconocer etiquetas
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Fuente	RUC - 3		
Descripción	La aplicación reconocerá el elemento con el que quiera interactuar el usuario mediante el uso de etiquetas impresas. Que identificarán de manera única a cada elemento.		
Cambios	Versión inicial: 01/04/2015		

Tabla 27: Requisito funcional 3 - Reconocer etiquetas

Requisito	F - 4	Nombre	Mostrar ayudas visuales
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Fuente	RUC - 4		
Descripción	La aplicación mostrará información útil como el estado o las operaciones a realizar cuando el usuario quiera interactuar con un elemento		
Cambios	Versión inicial: 01/04/2015		

Tabla 28: Requisito funcional 4 - Mostrar ayudas visuales

Requisito	F - 5	Nombre	Obtener comandos del servidor
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Fuente	RUC - 5		
Descripción	La aplicación recuperará los comandos disponibles del servidor, mostrando que acciones puede realizar.		
Cambios	Versión inicial: 01/04/2015		

Tabla 29: Requisito funcional 5 - Obtener comandos del servidor

Requisito	F - 6	Nombre	Definir ajustes
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Fuente	RUC - 6		
Descripción	La aplicación tendrá un menú de ajustes que permitirá definir parámetros de configuración.		
Cambios	Versión inicial: 01/04/2015		

Tabla 30: Requisito funcional 6 - Definir ajustes

Requisito	F - 7	Nombre	Ajustar dirección del servidor
Prioridad	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
Fuente	RUC - 7		

<b>Descripción</b>	La aplicación tendrá una sección de los ajustes para los datos del servidor.
<b>Cambios</b>	Versión inicial: 01/04/2015

Tabla 31: Requisito funcional 7 - Ajustar dirección del servidor

Requisito	F - 8	Nombre	Ajustar formato de información
<b>Prioridad</b>	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
<b>Fuente</b>	RUC - 8		
<b>Descripción</b>	La aplicación tendrá una sección de los ajustes para: <ul style="list-style-type: none"> <li>- Tamaño</li> <li>- Grosor</li> <li>- Color</li> <li>- Color de sombra</li> </ul>		
<b>Cambios</b>	Versión inicial: 01/04/2015		

Tabla 32: Requisito funcional 8 - Ajustar formato de información

Requisito	F - 9	Nombre	Aumentar zoom de video por voz
<b>Prioridad</b>	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
<b>Fuente</b>	RUC - 9		
<b>Descripción</b>	La aplicación tendrá un comando de voz para aumentar el zoom de la captura		
<b>Cambios</b>	Versión inicial: 01/04/2015		

Tabla 33: Requisito funcional 9 - Aumentar zoom de video por voz

Requisito	F - 10	Nombre	Disminuir zoom de video por voz
<b>Prioridad</b>	<input type="checkbox"/> Requerido	<input checked="" type="checkbox"/> Deseable	<input type="checkbox"/> Opcional
<b>Fuente</b>	RUC - 9		
<b>Descripción</b>	La aplicación tendrá un comando de voz para disminuir el zoom de la captura		
<b>Cambios</b>	Versión inicial: 01/04/2015		

Tabla 34: Requisito funcional 10 - Disminuir zoom de video por voz

### 3.3.3 Requisitos no funcionales

A continuación se especifican los requisitos no funcionales:

Requisito	NF - 1	Nombre	Funcionamiento con internet
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Fuente	RUR - 1		
Descripción	Para el funcionamiento de la aplicación será necesaria una conexión de internet.		
Cambios	Versión inicial: 01/04/2015		

Tabla 35: Requisito no funcional 1 - Funcionamiento con internet

Requisito	NF - 2	Nombre	Funcionamiento en Android
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Fuente	RUR - 2		
Descripción	La aplicación será desarrollada para su funcionamiento en un dispositivo móvil con Android 4.0 en adelante.		
Cambios	Versión inicial: 01/04/2015		

Tabla 36: Requisito no funcional 2 - Funcionamiento en Android

Requisito	NF - 3	Nombre	Funcionamiento en red local
Prioridad	[X] Requerido	[ ] Deseable	[ ] Opcional
Fuente	RUR - 3		
Descripción	La aplicación será capaz de comunicarse con un servidor que se encuentre en la misma red que el dispositivo que ejecute la aplicación		
Cambios	Versión inicial: 01/04/2015		

Tabla 37: Requisito no funcional 3 - Funcionamiento en red local

Requisito	NF - 4	Nombre	Funcionamiento con cámara
Prioridad	[X] Requerido    [ ] Deseable    [ ] Opcional		
Fuente	RUR - 4		
Descripción	La aplicación requerirá que el móvil donde se ejecute disponga de una cámara de video. La calidad de la captura repercutirá en la identificación de las etiquetas.		
Cambios	Versión inicial: 01/04/2015		

Tabla 38: Requisito no funcional 4 - Funcionamiento con cámara

Requisito	NF - 5	Nombre	Funcionamiento con microfono
Prioridad	[X] Requerido    [ ] Deseable    [ ] Opcional		
Fuente	RUR -5		
Descripción	La aplicación obtendrá los comandos mediante la escucha del micrófono		
Cambios	Versión inicial: 01/04/2015		

Tabla 39: Requisito no funcional 5 - Funcionamiento con micrófono

Requisito	NF - 6	Nombre	Funcionamiento con API de Google
Prioridad	[ ] Requerido    [X] Deseable    [ ] Opcional		
Fuente	RUR - 6		
Descripción	La aplicación hará uso del API de Google para el reconocimiento de voz, ofreciendo compatibilidad para futuras actualizaciones de Android.		
Cambios	Versión inicial: 01/04/2015		

Tabla 40: Requisito no funcional 6 - Funcionamiento con API de Google



Requisito	NF - 7	Nombre	Funcionamiento sin contacto
Prioridad	[ ] Requerido    [X] Deseable    [ ] Opcional		
Fuente	RUR - 7		
Descripción	El usuario no debe interactuar con la pantalla para llevar a cabo operaciones con los aparatos		
Cambios	Versión inicial: 01/04/2015		

Tabla 41: Requisito no funcional 7 - Funcionamiento sin contacto

Requisito	NF - 8	Nombre	Funcionamiento sin datos
Prioridad	[ ] Requerido    [X] Deseable    [ ] Opcional		
Fuente	RUR - 8		
Descripción	La información de las operaciones y dispositivos no se almacenará en la aplicación móvil, se obtendrán directamente del servidor local, de modo que no sea necesaria modificar la aplicación para añadir dispositivos u operaciones.		
Cambios	Versión inicial: 01/04/2015		

Tabla 42: Requisito no funcional 10 - Funcionamiento sin datos

### 3.3.4 Matriz de trazabilidad de requisitos

La matriz de trazabilidad muestra la correlación de los requisitos de software y los requisitos de usuario, de modo que se cumplan todos los requisitos de usuario.

Requisito Software Requisito Usuario	F - 1	F - 2	F - 3	F - 4	F - 5	F - 6	F - 7	F - 8	F - 9	F - 10	NF - 1	NF - 2	NF - 3	NF - 4	NF - 5	NF - 6	NF - 7	NF - 8
RUC - 1	■																	
RUC - 2		■																
RUC - 3			■															
RUC - 4				■														
RUC - 5					■													
RUC - 6						■												
RUC - 7							■											
RUC - 8								■										
RUC - 9									■									
RUR - 1											■							
RUR - 2												■						
RUR - 3													■					
RUR - 4														■				
RUR - 5															■			
RUR - 6																■		
RUR - 7																	■	
RUR - 8																		■

Tabla 43: Matriz de trazabilidad de requisitos

### 3.3.5 Casos de uso

Un caso de uso es un posible uso de la aplicación en una determinada situación llevada a cabo por el usuario. A continuación se definirán los posibles casos de uso:

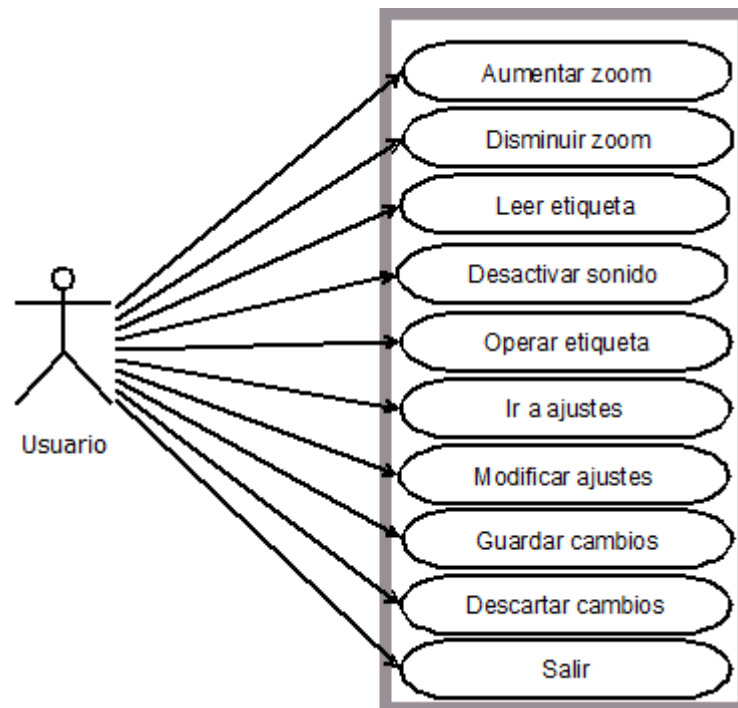


Ilustración 21: Diagrama de casos de uso

Aumentar el zoom de la visualización de video:

Identificador	C1
Nombre	Aumentar Zoom
Actor	Usuario
Objetivo	Aumentar el zoom de la captura de video
Precondiciones	Visualizar la captura de video y que el zoom no esté al máximo.
Postcondiciones	N / A
Escenario básico	1. Arrancar la aplicación 2. Esperar a que la captura de video y el reconocimiento de voz se inicialice. 3A. Pronunciar el comando de voz "Acercar". 3B. Pulsar el botón de acercar.

Tabla 44: Caso de uso 1 - Aumentar zoom

Disminuir el zoom de la visualización de video:

Identificador C2	
Nombre	Disminuir Zoom
Actor	Usuario
Objetivo	Disminuir el zoom de la captura de video
Precondiciones	Visualizar la captura de video y que el zoom no esté al mínimo.
Postcondiciones	N / A
Escenario básico	1. Arrancar la aplicación 2. Esperar a que la captura de video y el reconocimiento de voz se inicialice. 3A.Pronunciar el comando de voz "Alejar". 3B.Pulsar el botón de alejar.

Tabla 45: Caso de uso 2 - Disminuir zoom

Leer una etiqueta:

Identificador C4	
Nombre	Leer etiqueta
Actor	Usuario
Objetivo	Capturar una etiqueta e identificarla
Precondiciones	Visualizar la captura de video
Postcondiciones	N / A
Escenario básico	1. Arrancar la aplicación 2. Esperar a que la captura de video 3. Enfocar con la cámara de video a una etiqueta

Tabla 46: Caso de uso 4 – Leer etiqueta

Desactivar los mensajes del servidor:

Identificador C5	
Nombre	Desactivar sonido
Actor	Usuario
Objetivo	Desactivar los mensajes que mande el servidor
Precondiciones	Estar en la pantalla de visualización de video y que el sonido esté activado
Postcondiciones	N / A
Escenario básico	1. Arrancar la aplicación 2. Esperar a que la pantalla se inicialice. 3. Pulsar sobre el botón de sonido

Tabla 47: Caso de uso 5 – Desactivar sonido

Activar los sonidos del servidor:

Identificador C6	
Nombre	Activar sonido
Actor	Usuario
Objetivo	Activar los mensajes que mande el servidor
Precondiciones	Estar en la pantalla de visualización de video y que el sonido esté desactivado
Postcondiciones	N / A
Escenario básico	<ol style="list-style-type: none"> <li>1. Arrancar la aplicación</li> <li>2. Esperar a que la pantalla se inicialice.</li> <li>3. Pulsar sobre el botón de sonido</li> </ol>

Tabla 48: Caso de uso 6 – Activar sonido

Operar alguno de los comandos de una etiqueta:

Identificador C7	
Nombre	Operar etiqueta
Actor	Usuario
Objetivo	Mandar al servidor uno de los comandos de una etiqueta
Precondiciones	Estar en la pantalla de visualización de video y tener enfocada una etiqueta con comandos disponibles
Postcondiciones	N / A
Escenario básico	<ol style="list-style-type: none"> <li>1. Arrancar la aplicación</li> <li>2. Esperar a que la pantalla se inicialice</li> <li>3. Enfocar a una etiqueta</li> <li>4. Esperar a que el servidor ofrezca comandos a realizar</li> <li>5. Pronunciar uno de los comandos disponibles</li> </ol>

Tabla 49: Caso de uso 7 – Operar comando

Visualizar los ajustes de la aplicación:

Identificador C8	
Nombre	Visualizar ajustes
Actor	Usuario
Objetivo	Ver los ajustes almacenados en la aplicación.
Precondiciones	Estar en la pantalla de visualización de video
Postcondiciones	N / A
Escenario básico	<ol style="list-style-type: none"> <li>1. Arrancar la aplicación</li> <li>2. Esperar a que la pantalla y el reconocimiento de voz se inicialice.</li> <li>3A.Pronunciar el comando de voz “Ajustes”</li> <li>3B.Pulsar sobre el icono de ajustes</li> </ol>

Tabla 50: Caso de uso 8 – Visualizar ajustes

Modificar los ajustes de la aplicación

Identificador C9	
<b>Nombre</b>	Modificar ajustes
<b>Actor</b>	Usuario
<b>Objetivo</b>	Modificar los ajustes de la aplicación
<b>Precondiciones</b>	Estar en la pantalla de ajustes de la aplicación
<b>Postcondiciones</b>	N / A
<b>Escenario básico</b>	<ol style="list-style-type: none"> <li>1. Arrancar la aplicación</li> <li>2. Esperar a que la pantalla se inicialice.</li> <li>3. Acceder a la pantalla de ajustes.</li> <li>4. Modificar alguna de las propiedades de la pantalla.</li> <li>5. Pulsar sobre el botón de guardado.</li> </ol>

Tabla 51: Caso de uso 9 – Modificar ajustes

Descartar cambios en los ajustes de la aplicación:

Identificador C10	
<b>Nombre</b>	Descartar cambios
<b>Actor</b>	Usuario
<b>Objetivo</b>	Descartar los cambios hechos en la pantalla de ajustes
<b>Precondiciones</b>	Estar en la pantalla de ajustes de la aplicación
<b>Postcondiciones</b>	N / A
<b>Escenario básico</b>	<ol style="list-style-type: none"> <li>1. Arrancar la aplicación</li> <li>2. Esperar a que la pantalla se inicialice</li> <li>3. Acceder a la pantalla de ajustes</li> <li>4. Modificar los ajustes</li> <li>5. Pulsar sobre el botón de descartar cambios</li> </ol>

Tabla 52: Caso de uso 10 – Descartar cambios

Salir de la aplicación:

Identificador C11	
<b>Nombre</b>	Salir de la aplicación
<b>Actor</b>	Usuario
<b>Objetivo</b>	Salir de la aplicación
<b>Precondiciones</b>	Estar en la pantalla de visualización de video
<b>Postcondiciones</b>	N / A
<b>Escenario básico</b>	<ol style="list-style-type: none"> <li>1. Arrancar la aplicación</li> <li>2. Esperar a que la pantalla se inicialice</li> <li>3A. Pronunciar el comando de voz "Salir".</li> <li>3B. Pulsar sobre el botón de salida.</li> </ol>

Tabla 53: Caso de uso 11 – Salir

## 4. Diseño del Sistema

---

### 4.1 Introducción

Una vez definidos todos los requisitos del software, se procede a realizar el diseño de la aplicación. A continuación se detallarán las decisiones que se han tomado en cuanto al diseño.

### 4.2 Opciones de diseño

En los requisitos se definen a grandes rasgos las necesidades del cliente, pero deja a elección de los expertos los recursos a usar, a continuación se explicarán las opciones elegidas y los motivos.

#### 4.2.1 Representación de información

A la hora de mostrar información por pantalla se han estudiado 2 opciones. En primer lugar se ha hecho un prototipo utilizando ARToolkit, y como segunda opción se utilizó un prototipo haciendo uso de OpenCV.

##### 4.2.1.1 ARToolkit

El prototipo se ha hecho basándose en el proyecto AndAR, una pequeña aplicación en Android capaz de reconocer códigos de realidad aumentada y superponer figuras en el video capturado.

Mediante el prototipo se trató de evaluar el rendimiento así como la usabilidad de los códigos de realidad aumentada. Se logró hacer una aplicación capaz de reconocer códigos concretos y mostrar cubos de distintos colores y además se comprobó que se podía personalizar las figuras representadas haciendo el diseño con Blender y exportándolas para su uso en la aplicación.

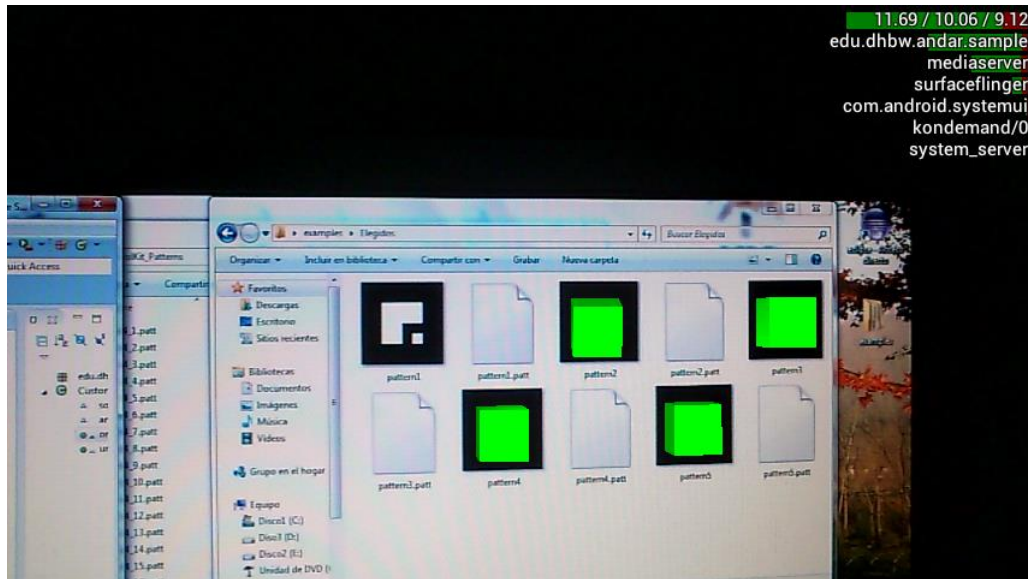


Ilustración 22: Prototipo AndAR. Captura 1

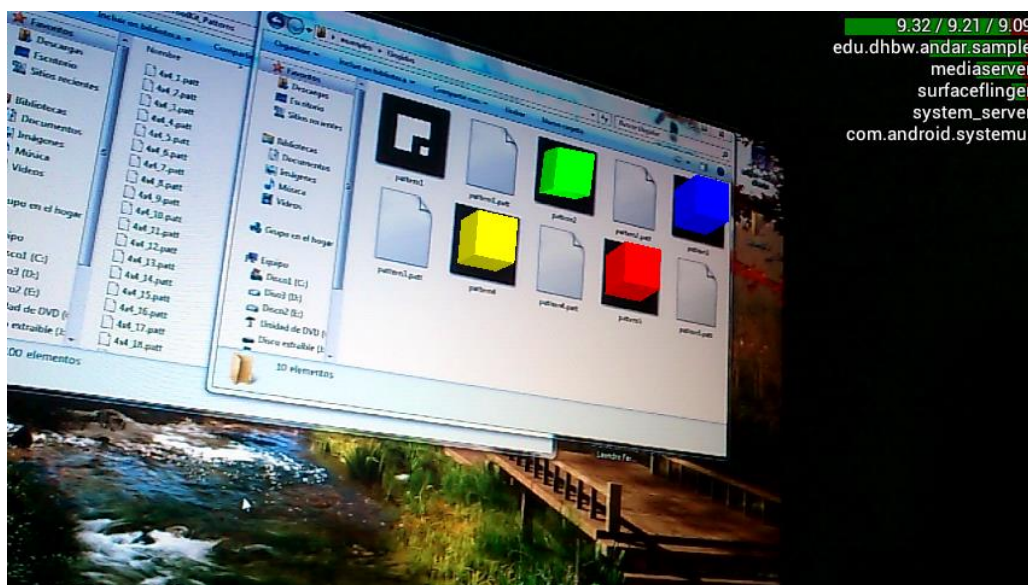


Ilustración 23: Prototipo AndAR. Captura 2

Con el prototipo se han comprobado aspectos positivos y negativos de su uso. Como cosas a favor, se ha probado que el rendimiento es bueno y el video obtenido a través de la aplicación ofrecía unos FPS bastantes altos, incluso para móviles de gama baja. Además, como se ha mencionado, ofrece la posibilidad de crear diseños con Blender y hacer uso de ellos en la aplicación. También cabe destacar, que los códigos son visibles desde una distancia a tener en cuenta.



Como aspectos negativos, se observó que su estabilidad no era la deseada, en ocasiones la aplicación finalizaba bruscamente y modificar la captura de video era altamente compleja, con lo cual no podía modificarse fácilmente para añadir información adicional a los objetos dibujados. Además, aunque los códigos se reconocen bastante bien a una distancia considerable, sacrifica el contener información compleja.

#### 4.2.1.2 OpenCV

Con el prototipo de OpenCV se valoró además la usabilidad de los códigos QR como marcador para la aplicación. Para ello, en el prototipo se hizo una funcionalidad para detectar códigos, los cuales se representaban con una cruz roja y sobre la figura, se dibujaba el contenido del código QR, el resultado fue el siguiente:

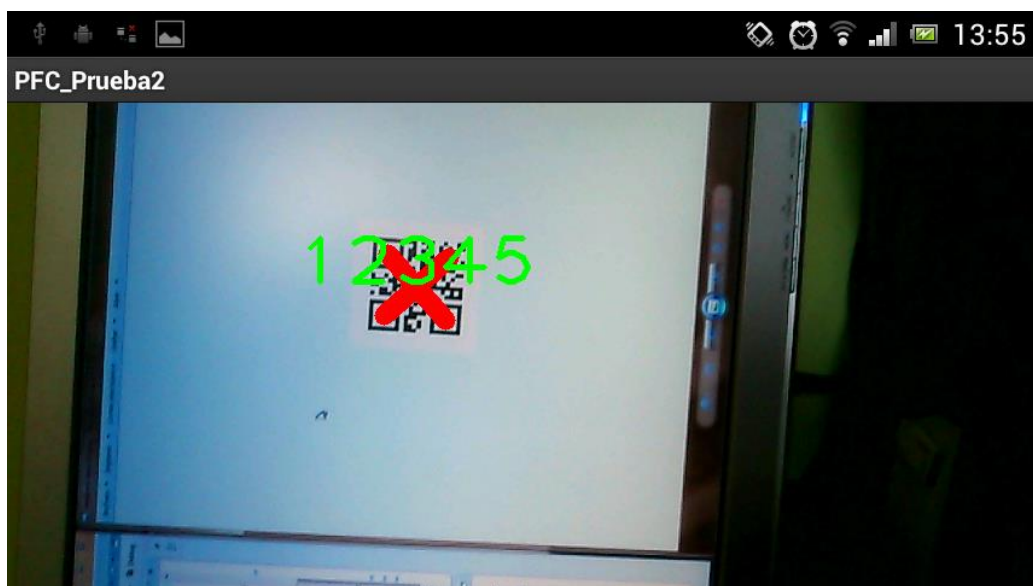


Ilustración 24: Prototipo OpenCV. Captura 1



Ilustración 25: Prototipo OpenCV. Captura 2

Como cosas a favor, se comprobó que la facilidad de uso de OpenCV para tratar el vídeo era mayor, pudiendo añadir información o modificar la imagen libremente, además, ofrece capturar el video tanto con el API de OpenCV como con la cámara nativa de Android, con lo cual permite aprovechar al máximo las prestaciones de la cámara del dispositivo. Además, se valoró muy positivamente el poder almacenar información codificada en el código QR.

Como puntos negativos, se encontró que el alcance de detección de estos marcadores es menor que con los códigos de realidad aumentada, sin embargo al poder usar el zoom de la cámara se compensa levemente. Además, aunque no nos ha parecido un gran problema, la librería ZXing con el que se hizo el prototipo, no era capaz de reconocer varios códigos en una misma captura.

#### 4.2.1.3 Solución adoptada

A continuación, se resume el análisis hecho de los dos prototipos y se concretan los puntos que se han valorado a la hora de escoger la tecnología a emplear, se pueden observar en la siguiente tabla:

Concepto	Mejor opción
Reconocimiento de marcadores	ARToolkit

<b>Fluidez de la aplicación</b>	ARToolkit
<b>Estabilidad de la aplicación</b>	OpenCV
<b>Tratamiento y edición de video</b>	OpenCV
<b>Versatilidad de los códigos</b>	OpenCV
<b>Facilidad de implementación</b>	OpenCV

Tabla 54: Valoración de prototipos

Con esta evaluación, se ha tomado la decisión de emplear para el desarrollo de la aplicación las librerías de OpenCV y los códigos QR.

#### 4.2.2 Reconocimiento de voz

Para la captura y reconocimiento de voz, se optado por emplear la utilidad que ofrece Android. Es una funcionalidad que trae por defecto desde la versión 2.2. Aunque en las últimas versiones ofrece su uso sin conexión a internet, descargando previamente el paquete de idiomas, se ha intentado hacer pruebas sin éxito, por lo tanto se ha definido como requisito la necesidad de internet. Sin embargo, al venir de fábrica, será muy difícil que deje de funcionar con los cambios de versión.

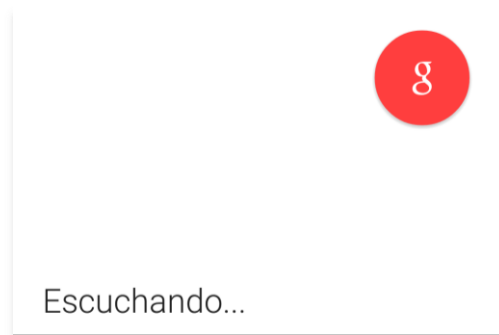


Ilustración 26: SpeechRecognizer de Google

### 4.3 Arquitectura del Sistema

En este apartado se va a intentar explicar la arquitectura definida en el desarrollo de la aplicación.

A continuación se puede observar el diagrama de la arquitectura:

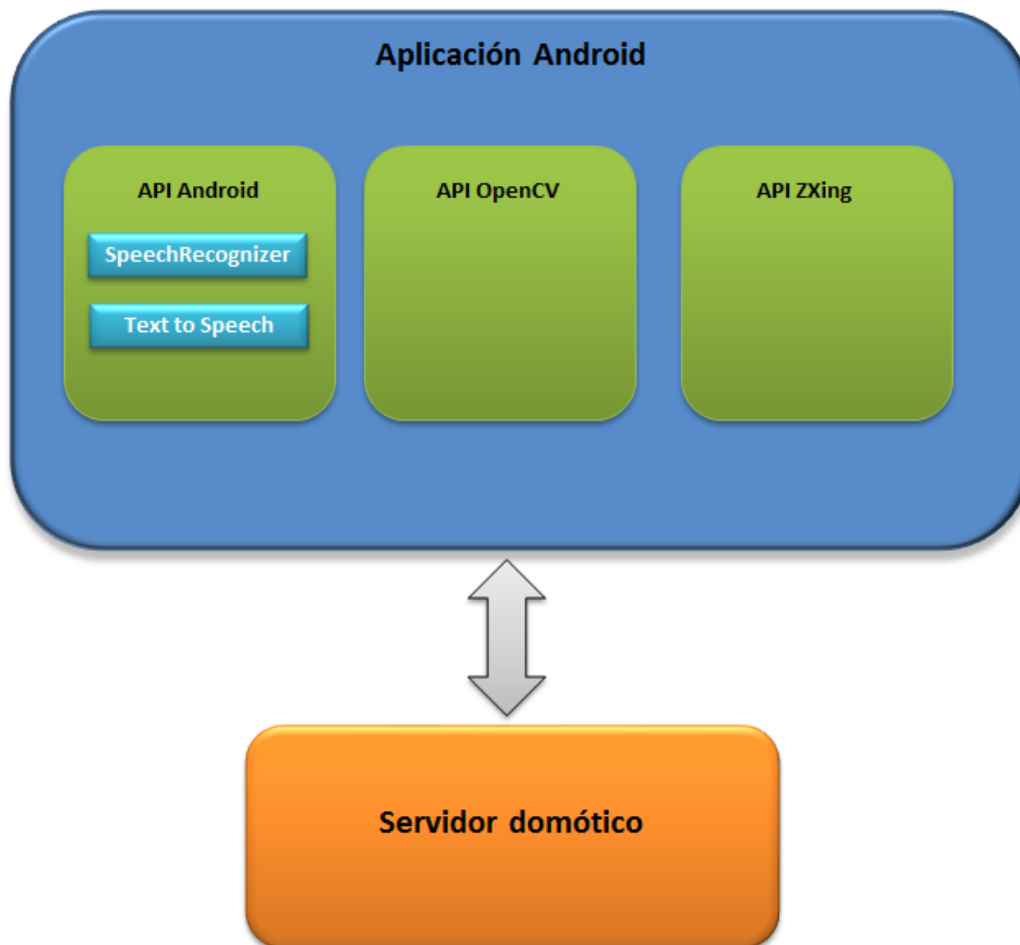


Ilustración 27: Arquitectura de la aplicación

## 4.4 Diseño UML del sistema

A continuación se expone el diagrama UML diseñado para el desarrollo del proyecto:

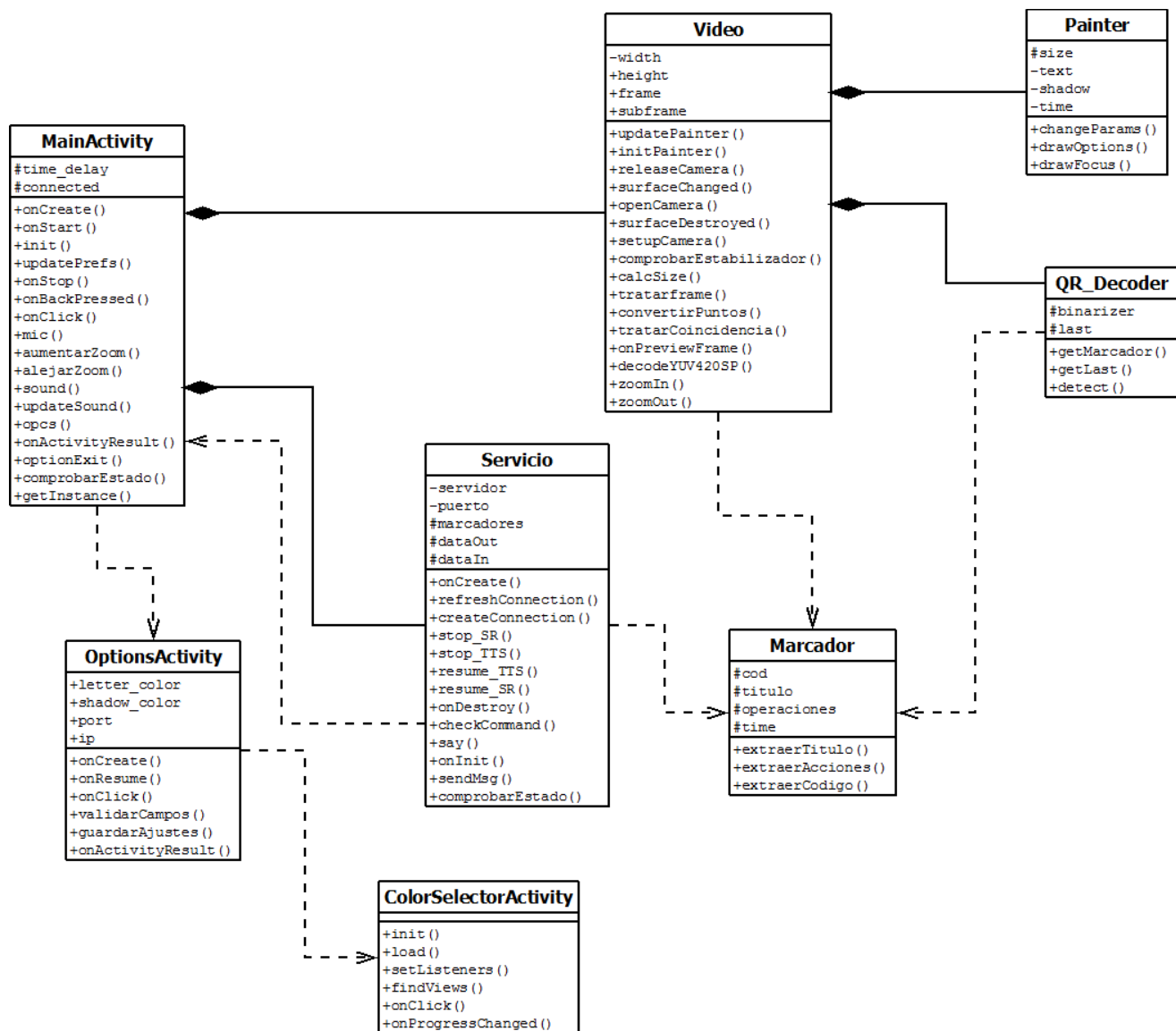


Ilustración 28: Diagrama de clases de la aplicación

## 4.5 Interfaz del sistema

A continuación, se exponen las interfaces que se han creado para las aplicaciones.

La pantalla principal, la cual muestra el video y la información recibida por el servidor, el área recuadrada es la zona de escaneo de códigos QR:

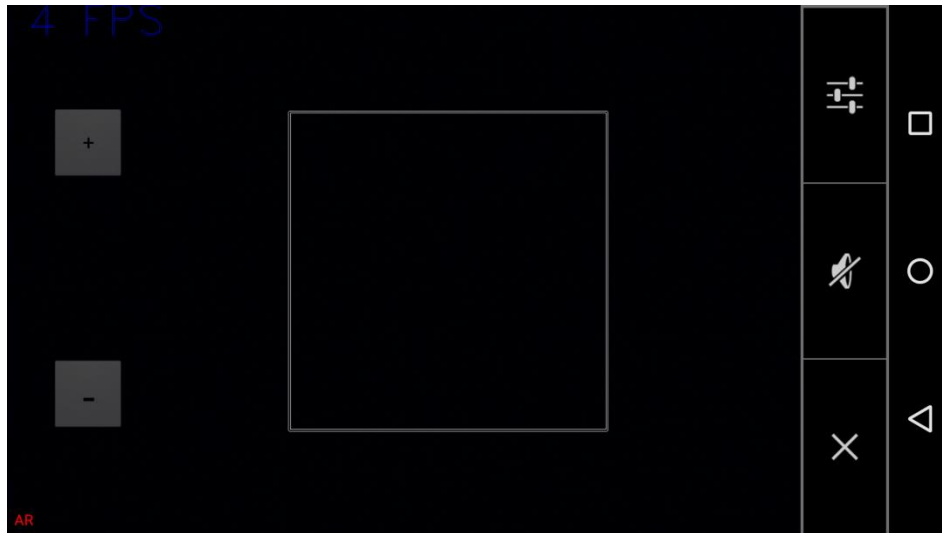


Ilustración 29: Pantalla principal de la aplicación

La pantalla de ajustes de la aplicación, a la cual se accede desde la pantalla principal:

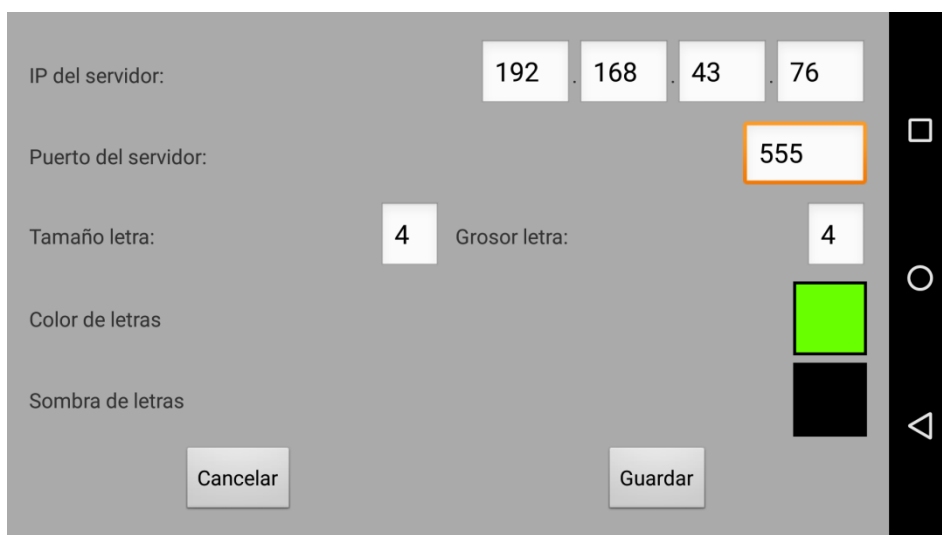


Ilustración 30: Pantalla de ajustes

Y la pantalla de selector de color, accesible desde el menú de ajustes:

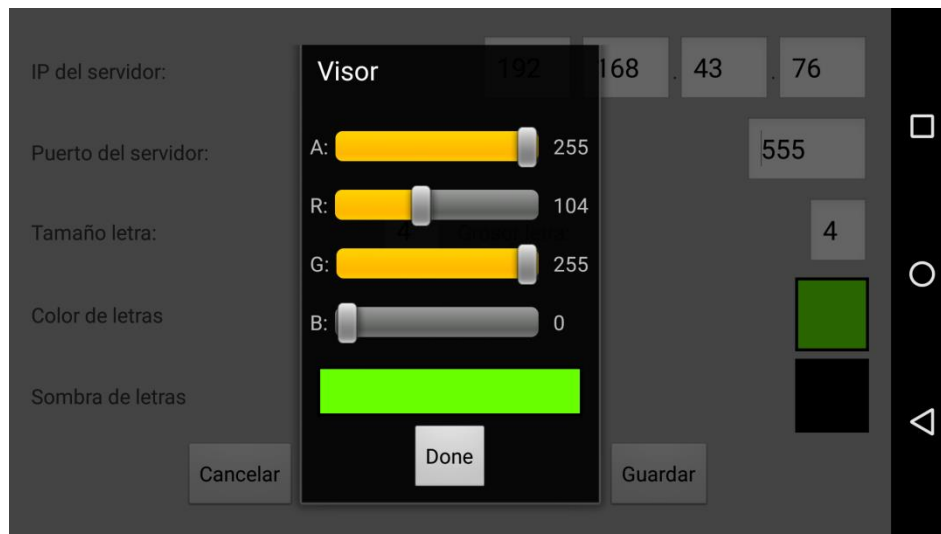


Ilustración 31: Pantalla de selector de color

### 4.2.3 Diagrama de navegación

A continuación, se muestra el diagrama de navegación de pantallas:

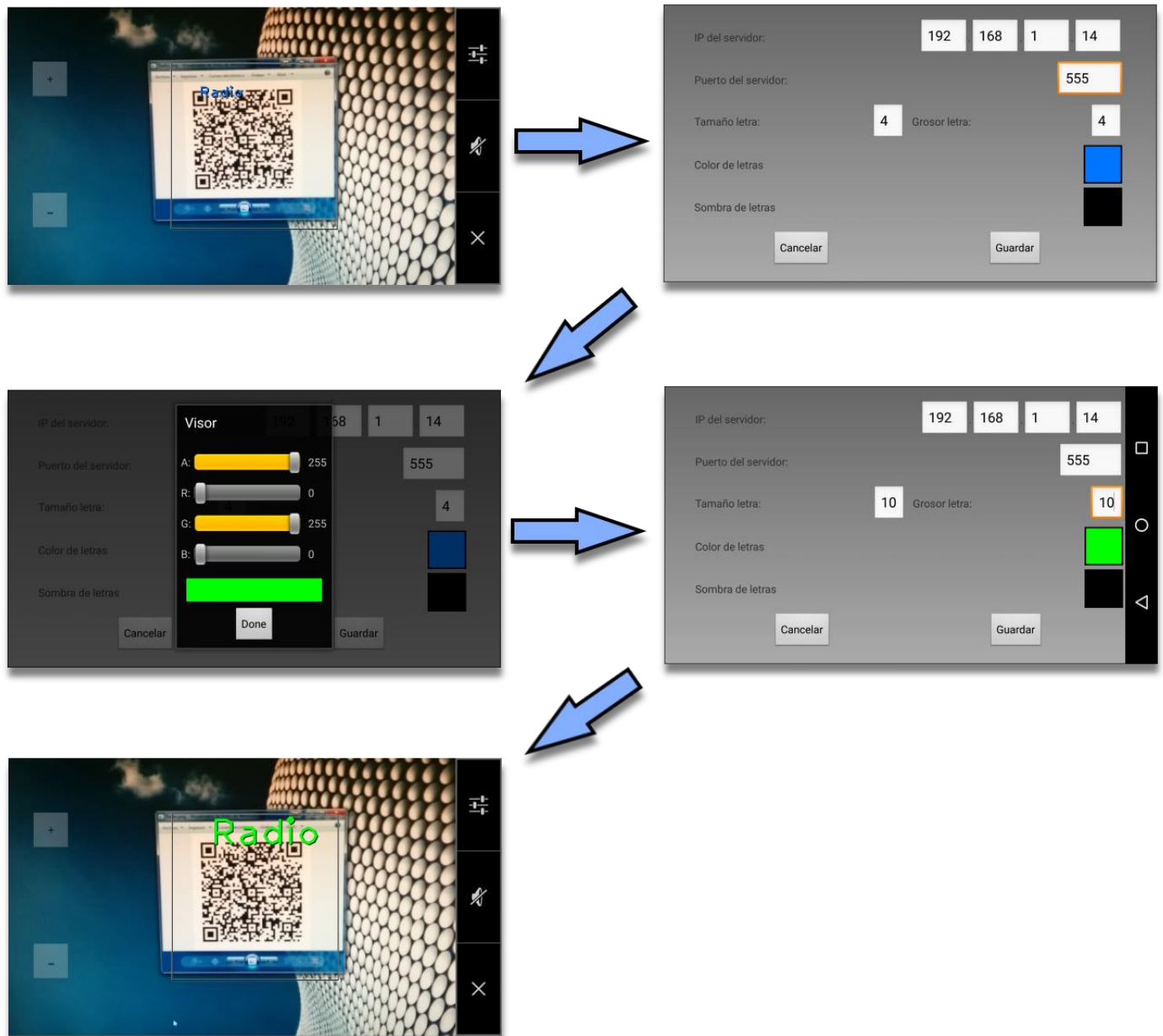


Ilustración 32: Diagrama de navegación



## 5. Implementación del Sistema

### 5.1 Introducción

En esta sección se va a dar más detalle sobre la implementación de las funcionalidades más importantes del desarrollo, así como las tecnologías utilizadas.

### 5.2 Herramienta de desarrollo del proyecto

Para el desarrollo de la aplicación se ha hecho uso de la herramienta Eclipse añadiendo el plugin ADT de integración con Android, el cual ha permitido crear proyectos Android y monitorizar dispositivos virtuales o los reales conectados al ordenador.

Esta herramienta ha cubierto todas las necesidades de desarrollo, sin embargo, desde la página para desarrolladores de Android, se informa de que van a finalizar el soporte oficial del plugin a finales del 2015 y recomiendan empezar a migrar los proyectos a su nueva herramienta de desarrollo Android Studio.


**Important:** Support for the Android Developer Tools (ADT) in Eclipse is ending, per our [announcement](#) . You should migrate your app development projects to Android Studio as soon as possible. For more information on transitioning to Android Studio, see [Migrating to Android Studio](#).

Ilustración 33: Fin del soporte a ADT

Además de Eclipse y el plugin ADT, ha sido necesario hacer uso del SDK de Android, esta herramienta permite la creación de dispositivos virtuales, y la descarga de los paquetes de cada una de las API's de Android así como más herramientas para desarrolladores Android.

La herramienta además da acceso a API's nuevas que todavía no han sido incluidas en ningún dispositivo Android:

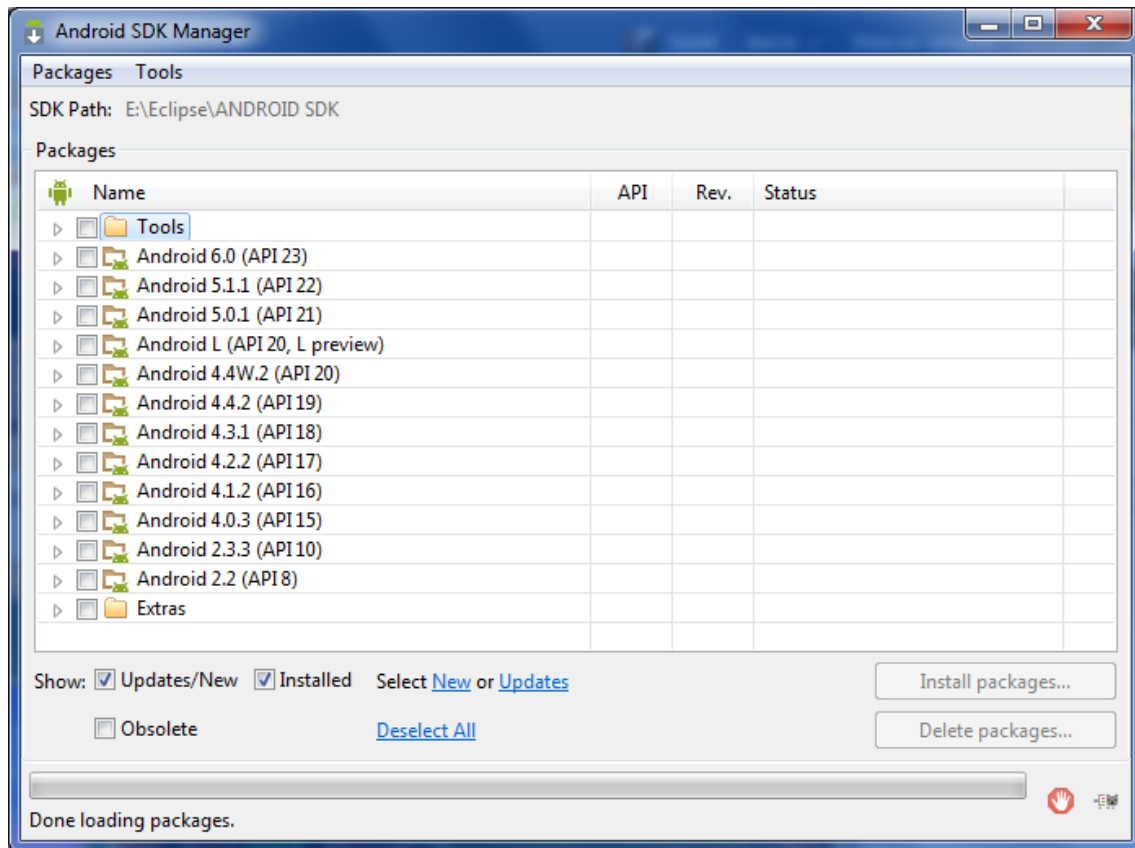


Ilustración 34: Android SDK Manager

## 5.3 Otros recursos

### 5.3.1 OpenCV

La librería de OpenCV tiene su propia aplicación en el Google Play Store que permite hacer uso de las librerías sin tener que incluirlas dentro de la aplicación:

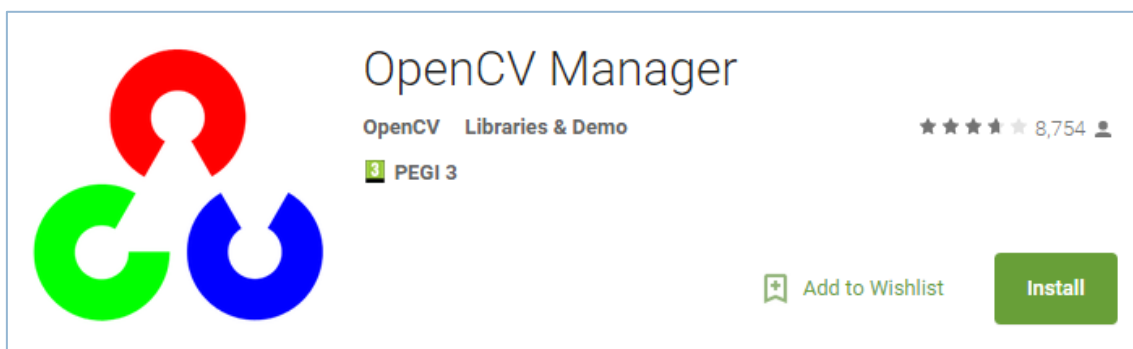


Ilustración 35: Aplicación en Google Play de OpenCV

Sin embargo, esto complica la instalación de aplicaciones que dependen de ella, además, la versión que está publicada no siempre es la última. Actualmente, la última versión disponible en la web de OpenCV es la 3.0 mientras que la que se incluye en la aplicación de Google Play es la 2.4.9, que está disponible en la web de OpenCV desde abril del 2014.

Por estos motivos, se ha optado por incluir las librerías dentro de la aplicación, aunque ello implica que no se actualizará de forma automática y que el proyecto va a ocupar más espacio.

### 5.3.2 ZXing

Las librerías de ZXing para el reconocimiento y lectura de los códigos QR están incluidas dentro del proyecto de la aplicación, se sacaron del proyecto oficial publicado en la página de GitHub de Zxing:

<https://github.com/zxing/zxing>

### 5.3.3 SpeechRecognizer

Para el reconocimiento de voz se ha utilizado el propio API de Android, concretamente la funcionalidad **SpeechRecognizer**. Android ofrece un *Intent* para reconocimiento de voz, lo cual inicia una actividad nueva y devuelve como resultado la cadena de texto identificada. En este caso, interesa realizar el reconocimiento de voz en segundo plano mientras se muestra por pantalla el video capturado.

Para realizar esto, se ha creado un servicio que implemente la interfaz **RecognitionListener** y **OnInitListener**, esto permite iniciar la escucha del micrófono en segundo plano e identificar lo que el usuario esté diciendo mientras visualiza el video.

### 5.3.4 Text To Speech

Además, se ha incluido la funcionalidad para que el servidor reproduzca mensajes hablados, para ello se ha hecho uso de la herramienta de Android Text To Speech. Esta herramienta es capaz de reproducir mensajes de voz en cualquiera de los idiomas instalados en Android, es un servicio gratuito que también está incluido en el API de Android.

## 5.4 Estructura del proyecto

Los ficheros de la aplicación se encuentran organizados siguiendo la estructura estándar de los proyectos Android, a excepción de la carpeta *src*, que contiene el código fuente de la aplicación y es propia de cada desarrollo:

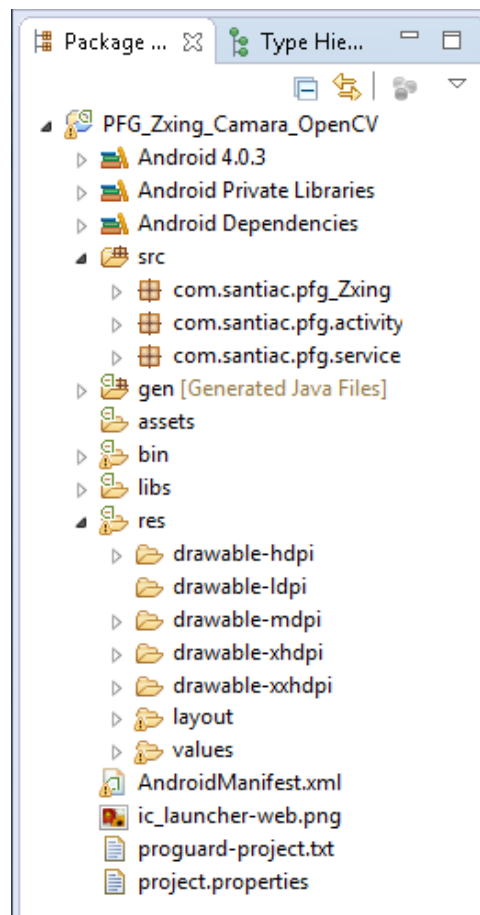


Ilustración 36: Estructura del proyecto

*src* : Contiene los ficheros .java de la aplicación, están clasificados en tres paquetes en función de su utilidad: el paquete *activity* contiene las clases que implementan las actividades, el paquete *service* contiene el servicio con la funcionalidades que se ejecutan en segundo plano, y el paquete *pfg\_Zxing* contiene toda la lógica de la captura de video y lectura de códigos.

*libs*: En esta carpeta se incluyen las librerías externas de la aplicación, en nuestro caso, están los ficheros de las librerías OpenCV y Zxing.

*res*: Esta carpeta contiene todos los recursos utilizados por la aplicación, entre los que se incluyen ficheros de layouts (uno por actividad), imágenes y ficheros de constantes XML.

## 6. Pruebas del Sistema

### 6.1 Introducción

En esta sección se detallarán las pruebas realizadas durante el desarrollo de la aplicación, cuyo fin es el de detectar cualquier fallo y asegurar que todo funcione correctamente.

### 6.2 Procedimiento de pruebas

La realización de las pruebas se ha realizado durante el desarrollo de la aplicación, fue necesario validar el correcto funcionamiento para poder avanzar con la planificación. A continuación se detallará las pruebas realizadas para cada tarea de desarrollo:

<b>IDENTIFICADOR</b>	<b>P – 01</b>
<b>Tarea asociada</b>	TD – 4
<b>Descripción</b>	Detección de código QR
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	Lectura en el log de Eclipse la detección del código QR
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Enfocar con la cámara del dispositivo a un código QR. -Validar que el código ha sido detectado.
<b>Resultado de la prueba</b>	OK

Tabla 55: Prueba funcional P – 01

<b>IDENTIFICADOR</b>	<b>P – 02</b>
<b>Tarea asociada</b>	TD – 4
<b>Descripción</b>	Lectura de contenido de código QR
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	Lectura en el log de Eclipse la detección del contenido del código QR
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Enfocar con la cámara del dispositivo a un código QR. -Validar que el código ha sido detectado. -Validar que el contenido del código QR se lee correctamente
<b>Resultado de la prueba</b>	OK

Tabla 56: Prueba funcional P – 02

<b>IDENTIFICADOR</b>	<b>P – 03</b>
<b>Tarea asociada</b>	TD – 4
<b>Descripción</b>	Impresión en pantalla del contenido del código QR
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	Lectura en la pantalla de la aplicación del contenido del código QR
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Enfocar con la cámara del dispositivo a un código QR. -Validar que el contenido del código se muestra por la pantalla.
<b>Resultado de la prueba</b>	OK

Tabla 57: Prueba funcional P – 03

<b>IDENTIFICADOR</b>	<b>P – 04</b>
<b>Tarea asociada</b>	TD – 5
<b>Descripción</b>	Visualización de la interfaz gráfica de la aplicación
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	Correcta visualización de la interfaz gráfica
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Validar que el contenido visualizado en la aplicación se corresponde con el diseño.
<b>Resultado de la prueba</b>	OK

Tabla 58: Prueba funcional P – 04

<b>IDENTIFICADOR</b>	<b>P – 05</b>
<b>Tarea asociada</b>	TD – 5
<b>Descripción</b>	Funcionalidad del botón salir
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	Aplicación finalizada
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Pulsar sobre el icono de salir. -Validar que se sale de la aplicación.
<b>Resultado de la prueba</b>	OK

Tabla 59: Prueba funcional P – 05

<b>IDENTIFICADOR</b>	<b>P – 06</b>
<b>Tarea asociada</b>	TD – 5
<b>Descripción</b>	Funcionalidad del botón sonido
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	Alternar icono de sonido activado/desactivado
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Pulsar sobre el icono de sonido. -Validar que el icono de sonido ha cambiado a deshabilitado. -Pulsar sobre el icono de sonido. -Validar que el icono de sonido ha vuelto a su estado original
<b>Resultado de la prueba</b>	OK

Tabla 60: Prueba funcional P – 06

<b>IDENTIFICADOR</b>	<b>P – 07</b>
<b>Tarea asociada</b>	TD – 6
<b>Descripción</b>	Captura de video de la resolución de la pantalla
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	La aplicación muestra la captura
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Validar en el log que la resolución del video capturado no excede la resolución de la pantalla.
<b>Resultado de la prueba</b>	OK

Tabla 61: Prueba funcional P – 07

<b>IDENTIFICADOR</b>	<b>P – 08</b>
<b>Tarea asociada</b>	TD – 6
<b>Descripción</b>	Aumento de zoom
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El zoom de la captura aumenta
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Pulsar sobre el botón [+]. -Validar que la imagen se ha acercado.
<b>Resultado de la prueba</b>	OK

Tabla 62: Prueba funcional P – 08

<b>IDENTIFICADOR</b>	<b>P – 09</b>
<b>Tarea asociada</b>	TD – 6
<b>Descripción</b>	Disminución de zoom
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El zoom de la captura disminuye
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Pulsar sobre el botón [-]. -Validar que la imagen se ha alejado.
<b>Resultado de la prueba</b>	OK

Tabla 63: Prueba funcional P – 09

<b>IDENTIFICADOR</b>	<b>P – 10</b>
<b>Tarea asociada</b>	TD – 7
<b>Descripción</b>	Reconocimiento de etiquetas codificadas
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	La aplicación muestra una lista de operaciones
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Enfocar a una etiqueta con un código QR codificado. -Validar que aparece el nombre del aparato y sus operaciones.
<b>Resultado de la prueba</b>	OK

Tabla 64: Prueba funcional P – 10

<b>IDENTIFICADOR</b>	<b>P – 10</b>
<b>Tarea asociada</b>	TD – 7
<b>Descripción</b>	Reconocimiento de etiquetas codificadas
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	La aplicación muestra una lista de operaciones
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Enfocar a una etiqueta con un código QR codificado. -Validar que aparece el nombre del aparato y sus operaciones.
<b>Resultado de la prueba</b>	OK

Tabla 65: Prueba funcional P – 10



<b>IDENTIFICADOR</b>	<b>P – 11</b>
<b>Tarea asociada</b>	TD – 8
<b>Descripción</b>	Reconocimiento de voz
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	La aplicación reconoce sonido
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Hablar de forma clara al dispositivo. -Validar que registra en el log lo que se ha dicho.
<b>Resultado de la prueba</b>	OK

Tabla 66: Prueba funcional P – 11

<b>IDENTIFICADOR</b>	<b>P – 12</b>
<b>Tarea asociada</b>	TD – 8
<b>Descripción</b>	Reconocimiento de comandos de voz Salir
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	La aplicación finaliza
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Decir de forma clara “Salir”. -Validar que la aplicación finaliza.
<b>Resultado de la prueba</b>	OK

Tabla 67: Prueba funcional P – 12

<b>IDENTIFICADOR</b>	<b>P – 13</b>
<b>Tarea asociada</b>	TD – 8
<b>Descripción</b>	Reconocimiento de comandos de voz Alejar
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El zoom disminuye
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Decir de forma clara “Alejar”. -Validar que el zoom de la captura disminuye.
<b>Resultado de la prueba</b>	OK

Tabla 68: Prueba funcional P – 13

<b>IDENTIFICADOR</b>	<b>P – 14</b>
<b>Tarea asociada</b>	TD – 8
<b>Descripción</b>	Reconocimiento de comandos de voz Acercar
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El zoom aumenta
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Decir de forma clara “Acercar”. -Validar que el zoom de la captura aumenta.
<b>Resultado de la prueba</b>	OK

Tabla 69: Prueba funcional P – 14

<b>IDENTIFICADOR</b>	<b>P – 15</b>
<b>Tarea asociada</b>	TD – 9
<b>Descripción</b>	Comunicación con servidor
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El servidor detecta la conexión de la aplicación
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Esperar a que se conecte al servidor. -Validar el servidor registra la conexión de la aplicación.
<b>Resultado de la prueba</b>	OK

Tabla 70: Prueba funcional P – 15

<b>IDENTIFICADOR</b>	<b>P – 16</b>
<b>Tarea asociada</b>	TD – 10
<b>Descripción</b>	Acceso al menú de ajustes
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El dispositivo muestra el menú de ajustes
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Pulsar sobre el botón de ajustes. -Validar que se muestra el menú.
<b>Resultado de la prueba</b>	OK

Tabla 71: Prueba funcional P – 16

<b>IDENTIFICADOR</b>	<b>P – 17</b>
<b>Tarea asociada</b>	TD – 10
<b>Descripción</b>	Modificación de color
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El dispositivo muestra el menú de selector de color y recupera el color
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Pulsar sobre el botón de ajustes. -Pulsar sobre la casilla de color. -Seleccionar un color. -Aceptar. -Validar que el color se ha actualizado.
<b>Resultado de la prueba</b>	OK

Tabla 72: Prueba funcional P – 17

<b>IDENTIFICADOR</b>	<b>P – 18</b>
<b>Tarea asociada</b>	TD – 10
<b>Descripción</b>	Guardado de ajustes al salir
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	La aplicación mantiene los cambios en los ajustes
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Pulsar sobre el botón de ajustes. -Modificar los ajustes. -Guardar cambios. -Salir de la aplicación. -Iniciar aplicación. -Pulsar sobre el botón de ajustes. -Validar que los ajustes se mantienen.
<b>Resultado de la prueba</b>	OK

Tabla 73: Prueba funcional P – 18

<b>IDENTIFICADOR</b>	<b>P – 19</b>
<b>Tarea asociada</b>	TD – 11
<b>Descripción</b>	Filtrado de las operaciones no disponibles
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El dispositivo muestra únicamente las

	operaciones disponibles
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Enfocar a una etiqueta con un código QR codificado con varias operaciones. -Validar que solo muestra las que ha respondido el servidor.
<b>Resultado de la prueba</b>	OK

Tabla 74: Prueba funcional P – 19

<b>IDENTIFICADOR</b>	<b>P – 20</b>
<b>Tarea asociada</b>	TD – 11
<b>Descripción</b>	Reconocimiento de operaciones del servidor
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El servidor recibe el comando de voz
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Enfocar a una etiqueta con un código QR codificado con varias operaciones. -Pronunciar de forma clara alguna de las operaciones mostradas mientras se enfoca al código. -Validar que la orden ha llegado al servidor.
<b>Resultado de la prueba</b>	OK

Tabla 75: Prueba funcional P – 20

<b>IDENTIFICADOR</b>	<b>P – 21</b>
<b>Tarea asociada</b>	TD – 11
<b>Descripción</b>	Reconocimiento de operaciones sin ver marcador
<b>Estado inicial</b>	Aplicación instalada en el dispositivo móvil
<b>Estado final</b>	El servidor recibe el comando de voz
<b>Procedimiento de prueba</b>	-Iniciar aplicación. -Enfocar a una etiqueta con un código QR codificado con varias operaciones. -Dejar de enfocar al marcador. -Pronunciar una de las operaciones antes de 1.5 segundos. -Validar que el comando de voz ha llegado al servidor.
<b>Resultado de la prueba</b>	OK

Tabla 76: Prueba funcional P – 19

## 7. Gestión del Proyecto

---

A continuación se detallará la gestión, análisis y desarrollo del proyecto, en sus diferentes fases y se explicarán las decisiones adoptadas.

### 7.1 Introducción

Se intentará realizar una buena planificación del proyecto, para ello, se hará un análisis preliminar de las necesidades así como de los recursos disponibles para llevarlo a cabo. Una vez evaluadas las posibilidades, se detallará y planificará la solución escogida.

### 7.2 Metodología del proyecto

Dadas las características del proyecto se ha optado por utilizar una metodología ágil, que permita obtener entregas tempranas de resultados tangibles y que permita hacer cambios durante el transcurso del proyecto en función de los resultados obtenidos. Por ello, la realización del proyecto se ha llevado a cabo mediante la metodología ágil Scrum.

Esta metodología se caracteriza por seguir una estrategia de desarrollo incremental, en lugar de utilizar una planificación y ejecución completa del producto.

Al tratarse de un Proyecto de Fin de Grado, y ser realizado por una única persona, no se llevará a cabo de forma íntegra, adaptándola a las necesidades.

### 7.3 Definición de roles.

#### 7.3.1 Propietario del producto.

El propietario del producto (product owner) es la persona encargada de tomar las decisiones del cliente. Para simplificar la comunicación y toma de decisiones, es necesario que este rol recaiga en una única persona. En el caso de que sea una organización, como puede ser la Universidad Carlos III de Madrid, las decisiones recaerán en el Tutor del PFG, Javier Fernández Muñoz.

Su cometido será decidir cómo será el resultado final, y el orden en que se irá construyendo los sucesivos incrementos: que se pone y que se quita de la pila del producto y cuál es la prioridad de las funcionalidades.

Según las circunstancias, es posible incluso que delegue en el equipo de desarrollo, o en alguien de su confianza, pero la responsabilidad será suya.

### 7.3.2 Equipo de desarrollo

Al tratarse de un proyecto de fin de grado, el equipo de desarrollo estaría formado por los posibles alumnos que estuvieran realizando el proyecto, en este caso al ser individual, no se cumple la recomendación se entre 3 y 8 personas y constará de una sola persona, Santiago de Arriba Cortijo.

## 7.4 Planificación inicial

Para la planificación de tareas se va a detallar una tabla en la que figuren cada una de las etapas de desarrollo y otra en la que se detallen las tareas de implementación y pruebas. Para la medición de las tareas se va a utilizar como medida de tiempo horas de trabajo de una persona.

### 7.4.1 Estimación y planificación de las tareas

Las tareas de todo el desarrollo se detallan en la siguiente tabla, dejando aparte las subtareas para la implementación y pruebas:

Tarea	Descripción	Clave
Evaluación	Evaluación del sistema que se quiere desarrollar	T – 1
Análisis	Análisis del sistema a desarrollar	T – 2
	Obtención de requisitos	T – 2A
	Definición de casos de uso	T – 2B
Gestión	Definir la gestión del proceso de desarrollo	T – 3
	Definir los pasos para la gestión de las tareas	T – 3A
	Elaborar la planificación inicial	T – 3B
	Elaborar el presupuesto	T – 3C
Diseño	Especificar el diseño de la aplicación	T – 4
	Diseñar la arquitectura de la aplicación	T – 4A
	Diseñar la interfaz de la aplicación	T – 4B
	Diseñar patrón para etiquetas con código QR	T – AC
Implementación / Pruebas	Se detalla en la siguiente tabla	T – 5

Tabla 77: Tareas del proyecto

De los requisitos extraídos en la fase de análisis se va elaborar la siguiente tabla de tareas a completar para terminar la etapa de implementación y pruebas:

Tarea	Descripción	Clave	Requisitos
Preparación de entorno	Instalación del software necesario para el desarrollo	TD – 1	
Prototipo ARToolkit	Creación de un prototipo usando ARToolkit (AndAR)	TD – 2	F - 1 NF -2
Prototipo OpenCV	Creación de un prototipo usando OpenCV	TD – 3	F - 1 NF -2
Lectura de códigos QR	Implementar funcionalidad para lectura de códigos QR en tiempo real	TD – 4	F - 3
Creación interfaz	Creación de la interfaz gráfica de la pantalla de captura de video	TD – 5	F – 1
Adaptación de captura y zoom	Mejorar el rendimiento de la aplicación limitando la resolución de captura a la resolución del dispositivo, con posibilidad de hacer zoom	TD – 6	F – 9 F – 10
Impresión de información	Implementar funcionalidad para mostrar información sobre las etiquetas	TD – 7	F – 4
Captura de comandos	Implementar la funcionalidad de captura de sonidos.	TD – 8	F - 2
Comunicación con servidor	Implementar la funcionalidad para comunicarse con el servidor	TD – 9	F – 5
Creación de interfaz ajustes	Creación de la interfaz de la pantalla de ajustes	TD – 10	F – 6 F – 7 F – 8
Creación de servidor de pruebas	Creación de un servidor para pruebas de integración	TD – 11	F – 5

Tabla 78: Tareas de implementación y pruebas

Con la información de las tablas anteriores ya podemos realizar una estimación de horas para cada tarea y fijar una fecha de inicio y de fin estimada. Con esta información podremos realizar un presupuesto y evaluar posibles retrasos durante el avance del proyecto.

Para cada una de las tareas definidas se han definido las siguientes fechas y estimación de horas:

Clave	Tarea	Fecha inicio	Fecha fin	Estimación en horas
T – 1	<b>Evaluación</b>	23/03/2015	27/03/2015	4 h
T – 2	<b>Análisis</b>	30/03/2015	03/04/2015	8 h
T – 2A	Requisitos	30/03/2015	01/04/2015	6 h
T – 2B	Casos de uso	02/04/2015	03/04/2015	2 h
T – 3	<b>Gestión</b>	06/04/2015	17/04/2015	16 h
T – 3A	Gestión de tareas	06/04/2015	11/04/2015	8 h
T – 3B	Planificación	14/04/2015	16/04/2015	6 h
T – 3C	Presupuesto	17/04/2015	17/04/2015	2 h
T – 4	<b>Diseño</b>	20/04/2015	01/05/2015	16 h
T – 4A	Arquitectura	20/04/2015	23/04/2015	6 h
T – 4B	Interfaz	24/04/2015	28/04/2015	6 h
T – AC	Etiquetas	29/04/2015	01/05/2015	4 h
T – 5	<b>Implementación / Pruebas</b>	04/05/2015	30/06/2015	168 h
	<b>Total</b>	23/03/2015	30/06/2015	212 h

Tabla 79: Planificación y estimación del proyecto

Para las tareas de implementación y pruebas se define la siguiente estimación:

Clave	Título	Fecha inicio	Fecha fin	Estimación en horas
TD – 1	<b>Preparación entorno</b>	04/05/2015	04/05/2015	4 h
TD – 2	<b>Prototipo ARToolkit</b>	05/05/2015	12/05/2015	24 h
TD – 3	<b>Prototipo OpenCV</b>	13/05/2015	20/05/2015	24 h
TD – 4	<b>Lectura de códigos</b>	21/05/2015	22/05/2015	8 h
TD – 5	<b>Creación interfaz</b>	25/05/2015	28/05/2015	16 h
TD – 6	<b>Adaptación captura</b>	29/05/2015	02/06/2015	12 h
TD – 7	<b>Impresión información</b>	03/06/2015	08/06/2015	16 h
TD – 8	<b>Captura de comandos</b>	09/06/2015	15/06/2015	20 h
TD – 9	<b>Comunicación servidor</b>	16/06/2015	19/06/2015	16 h
TD – 10	<b>Creación ajustes</b>	22/06/2015	24/06/2015	12 h
TD – 11	<b>Servidor de pruebas</b>	25/06/2015	30/06/2015	16 h

Tabla 80: Planificación de implementación y pruebas



## 7.4.2 Diagrama de Gantt

A continuación se representa de manera gráfica la planificación de las diferentes etapas del proyecto y como se distribuyen en el tiempo.

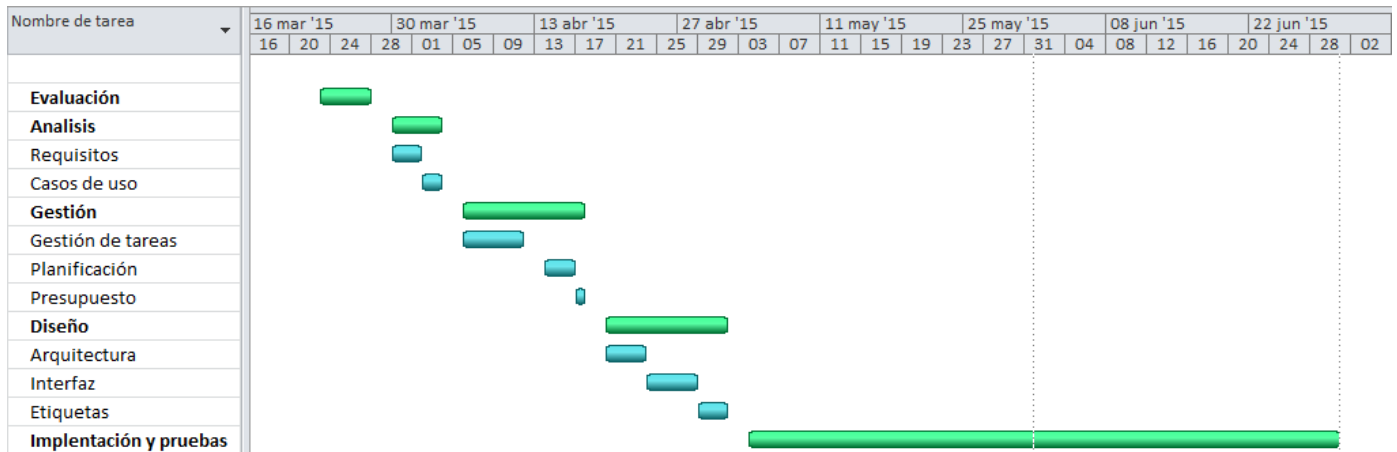


Ilustración 37: Diagrama de Gantt

## 7.5 Presupuesto del proyecto

En este apartado se detalla el presupuesto del proyecto, especificando los diferentes costes derivados de su desarrollo:

PRESUPUESTO DE PROYECTO					
Autor					
Santiago de Arriba Cortijo					
Departamento					
Area de aplicaciones móviles					
Descripción del Proyecto					
-Título	Diseño e implementación de una interfaz de domótica asistencial basado en realidad aumentada				
-Duración (meses)	3				
IVA	21%				
Presupuesto total del Proyecto					
4449,30 Euros					
Desglose del presupuesto					
Personal					
Apellidos y nombre	Tarea	Dedicación (horas)	Coste hombre mes	Coste (Euros)	
de Arriba Cortijo, Santiago	Gestión de proyecto	16	18,83	301,28	
	Análisis y diseño	28	16,55	463,4	
	Programación	168	14,61	2454,48	
	Total	212		3219,16	
Equipos					
Equipos	Coste €	% Uso dedicado proyecto	Dedicación (meses)	Periodo de depreciación (meses)	Coste imputable *
Ordenador portátil	1199	100	3	60	59,95
Teléfono móvil Nexus 5	399	100	3	36	33,25
				Total	93,20
*Cálculo de la amortización		= C/P*D*U			
		C - Coste del producto P - Periodo de depreciación D - Cantidad de uso U - % de utilización			
Otros Costes del proyecto					

Descripción		Unidades	Coste Unidad	Coste Total
Licencia Microsoft Office Hogar y Empresas 2013		1	269	269
Licencia Microsoft Windows 8.1		1	95,75	95,75
			Total	364,75
Resumen de costes				
Presupuesto Costes totales	Coste en €			
Personal	3219,16			
Amortización	93,20			
Costes de funcionamiento	364,75			
Total sin IVA	3677,11			
IVA	772,19			
Total	4449,30			

Tabla 81: Presupuesto del proyecto

## 7.6 Planificación final

Con el objetivo de evaluar la planificación efectuada, debemos comparar si el tiempo estimado se ha desviado en exceso con respecto al tiempo que finalmente se ha necesitado para llevar a cabo el proyecto.

Para ello vamos a contrastar en primer lugar las fechas de las tareas, de este modo se sabrá si se ha entregado a tiempo. A continuación, podemos ver una tabla comparativa con las fechas de inicio y fin iniciales, junto con los datos reales y su desviación:

Tarea	Estimada		Real		Desviación
	Fecha Inicio	Fecha fin	Fecha Inicio	Fecha fin	
<b>Evaluación</b>	23/03/2015	27/03/2015	20/03/2015	27/03/2015	
<b>Análisis</b>	30/03/2015	03/04/2015	30/03/2015	05/04/2015	+2
Requisitos	30/03/2015	01/04/2015	30/03/2015	01/04/2015	
Casos de uso	02/04/2015	03/04/2015	02/04/2015	05/04/2015	
<b>Gestión</b>	06/04/2015	17/04/2015	06/04/2015	20/04/2015	
Gestión de tareas	06/04/2015	11/04/2015	06/04/2015	11/04/2015	
Planificación	14/04/2015	16/04/2015	14/04/2015	16/04/2015	
Presupuesto	17/04/2015	17/04/2015	17/04/2015	17/04/2015	
<b>Diseño</b>	20/04/2015	01/05/2015	20/04/2015	30/04/2015	-1
Arquitectura	20/04/2015	23/04/2015	20/04/2015	23/04/2015	
Interfaz	24/04/2015	28/04/2015	24/04/2015	28/04/2015	

<b>Etiquetas</b>	29/04/2015	01/05/2015	29/04/2015	30/04/2015	
<b>Implementación / Pruebas</b>	04/05/2015	30/06/2015	04/05/2015	05/07/2015	+5
<b>Total</b>	23/03/2015	30/06/2015	23/03/2015	05/07/2015	

Tabla 82: Comparativa de fechas estimadas y reales

Como puede verse en la tabla, ha habido una desviación de una semana respecto a la estimación inicial, la mayoría se corresponde con la fase de implementación y pruebas, la cual ha tenido imprevistos durante las pruebas que han alargado la fase.

A continuación se va a comparar las horas que han llevado realizar cada una de las fases, es importante puesto que repercute directamente en los beneficios del proyecto:

<b>Tarea</b>	<b>Estimación en horas</b>	<b>Tiempo real en horas</b>	<b>Desviación</b>
<b>Evaluación</b>	<b>4 h</b>	<b>6 h</b>	<b>+2</b>
<b>Análisis</b>	<b>8 h</b>	<b>14 h</b>	<b>+6</b>
Requisitos	6 h	10 h	+4
Casos de uso	2 h	4 h	+2
<b>Gestión</b>	<b>16 h</b>	<b>13 h</b>	<b>-3</b>
Gestión de tareas	8 h	6 h	-2
Planificación	6 h	5 h	-1
Presupuesto	2 h	2 h	
<b>Diseño</b>	<b>16 h</b>	<b>12 h</b>	<b>-4</b>
Arquitectura	6 h	4 h	-2
Interfaz	6 h	6 h	
Etiquetas	4 h	2 h	-2
<b>Implementación / Pruebas</b>	<b>168 h</b>	<b>180 h</b>	<b>+12</b>
<b>Total</b>	<b>212 h</b>	<b>225 h</b>	<b>+13</b>

Tabla 83: Comparativa de horas estimadas y reales

Como se puede observar, el balance final ha sido una desviación de 13 horas, las cuales están repartidas en varias fases del proyecto. Esta desviación, es negativa en algunas fases y positivas en otras, por lo tanto el cálculo del coste será  $66,20 - 56,49 + 175,32 = 185,03$  €, que restará beneficios al proyecto.

## 8. Conclusiones

---

En este apartado se van a explicar las conclusiones del proyecto, además de las personales, derivadas de la gestión y de la implementación del mismo.

### 8.1 Conclusiones del proyecto

Con la realización de este proyecto se logrado diseñar e implementar una interfaz de domótica asistencial basada en realidad aumentada, cumpliendo a su vez, los objetivos secundarios que se habían propuesto:

- Se ha logrado crear una solución para un dispositivo móvil que permita obtener información del estado de los aparatos conectados en el hogar.
- Se ha conseguido interactuar mediante el uso de comandos de voz para operar desde el dispositivo móvil.
- Se ha investigado técnicas para la creación de aplicaciones de realidad aumentada.
- Se ha creado una aplicación que pueda ser usada y servir de ayuda a personas de movilidad reducida.

### 8.2 Conclusiones personales

La realización de este proyecto me ha permitido adquirir conocimientos acerca de la gestión de proyectos, de la investigación y la búsqueda de información:

- Una buena gestión del proyecto, especialmente la planificación es importante, hace que el proyecto sea más eficiente y ahorra costes derivados de su prolongación en el tiempo.
- Tenía conocimientos previos en Android, pero nunca me había planteado utilizar realidad aumentada en mis proyectos. Me ha supuesto un reto personal y me ha llevado a tener que investigar bastante para solucionar los problemas que iba encontrando. Además, me ha servido para darme cuenta de que la realidad aumentada tiene más usos de los que conocía, haciéndome ver que hay más aplicaciones de en el mercado, algunas de las cuales han ido surgiendo mientras desarrollaba el proyecto.

- Android ofrece muchas posibilidades, pero tiene un API de programación muy fragmentado. Obliga a decidir entre utilizar las novedades de las últimas versiones, limitando con ello el número de dispositivos que podrán usar tu aplicación, o por el contrario, no hacer uso de ellas y de ese modo hacer tu aplicación compatible con un mayor número de dispositivos.
- La utilización de ciertas API's de Google te dan confianza en el sentido de que van a ser compatibles con versiones posteriores, sin embargo puede ser que modifiquen sus modos de uso, provocando que tengas que adaptar la aplicación que funcionaba correctamente en versiones anteriores.

### 8.3 Trabajos futuros

Como propuestas para líneas futuras de este proyecto, a continuación se ofrecen una serie de puntos que podrían ser mejorados en futuras versiones:

- ✚ La utilización del API de Google para la captura de sonido ha cumplido nuestras necesidades, pero requiere de una conexión a internet. Si bien es cierto que actualmente la mayoría de hogares disponen de conexión ADSL o de fibra óptica, una posible mejora sería sustituir dicho API por una solución que no la requiera, ya que ante un corte en dicho servicio, la aplicación dejaría de funcionar.
- ✚ El diseño de la aplicación ha quedado un poco desfasada con respecto a las líneas de diseño que aconseja Google, una posible línea de mejora sería rediseñar el menú de ajustes haciendo uso de *Material Design*.
- ✚ Aunque Android es el sistema operativo móvil más extendido, la aplicación podría ser portada a iOS o Windows Phone con el objetivo de llegar a más público.

## ANEXO I – Manual usuario de la aplicación

Este anexo tiene como objetivo explicar de manera simple la utilización de la aplicación.

La aplicación consta de 3 pantallas, las cuales se explicarán en detalle:

**-Pantalla de captura de video.** Desde esta pantalla se pueden ejecutar los comandos de voz. Para poder ejecutarlos, tienen que ser reconocidos previamente por la aplicación, aunque se deje de ver la etiqueta, se dispone de 5 segundos o hasta que se lea otra etiqueta, para pronunciar alguno de los comandos de voz que hayan salido por pantalla, pasado ese tiempo la lectura del código expirará.

Para aumentar el rendimiento de la lectura de código se ha delimitado el área de escaneo al cuadrado que sirve de guía.

Además de los comandos de las etiquetas, existen comandos de voz reservados, que no se podrán usar como comandos, ya que no serán reconocidos como operaciones, estos comandos son:

- **Acercar.** Aumenta el zoom de la captura.
- **Alejar.** Disminuye el zoom de la captura.
- **Ajustes.** Accede al menú de ajustes.
- **Salir.** Sale de la aplicación.



Ilustración 38: Pantalla de captura de video

**-Pantalla de ajustes.** Desde esta pantalla se puede personalizar el texto de la pantalla de captura de video y modificar los parámetros del servidor. Los distintos ajustes son:

- **IP del servidor.** Define la dirección física del servidor, esta debe estar en la misma red local que el dispositivo móvil.

- Puerto del servidor.** Define el puerto de escucha del servidor.
- Tamaño letra.** Permite aumentar o disminuir el tamaño de las letras.
- Grosor letra.** Permite aumentar o disminuir el grosor de las letras.
- Color de letras.** Define el color principal de las letras.
- Sombra de letras.** Define el color secundario de las letras.

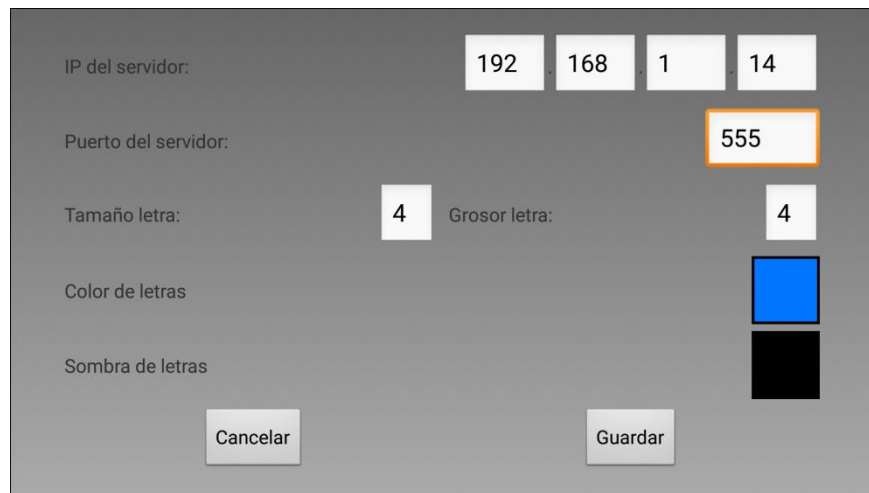
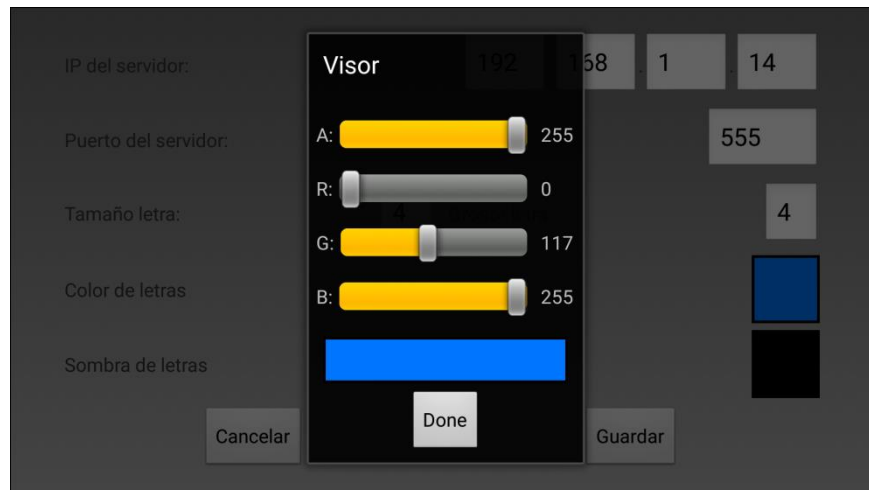


Ilustración 39: Pantalla de ajustes

-**Pantalla de selector de color.** Esta pantalla permite personalizar los colores de las letras o sus sombras, la pantalla para ambos ajustes es la misma. Para una mayor personalización, se define el color por 4 propiedades:

- A.** Define la claridad del color.
- R.** Define el componente rojo.
- G.** Define el componente verde.
- B** Define el componente azul.





**Ilustración 40: Pantalla de selector de color**

## ANEXO II – Manual de creación de marcadores

Esta sección está destinada a explicar el esquema definido en los códigos QR para poder ser utilizados por la aplicación.

La aplicación ha sido diseñada para que la información acerca de los aparatos, así como sus operaciones, sea leída de los propios códigos y no esté guardada en la aplicación, de este modo, no será necesario alterar la aplicación para añadir aparatos u operaciones.

El esquema definido, y el cual debe ser respetado para la correcta lectura de los códigos está formado por tres secciones, las 3 deben estar presentes y deben respetar los nombres de las secciones:

- **Título:** Será el nombre que aparezca por pantalla a la hora de mostrar la información del código, se delimita por los tags <TITULO> y <\TITULO>.
- **Acciones:** Serán las operaciones que podrán realizarse sobre el aparato, únicamente se mostrarán aquellas que el servidor considere que pueden llevarse a cabo. La cardinalidad debe ser de 1 a n y está compuesto por el nombre de la acción y el código asociado a la acción, en ese orden y separados por ::, en el caso de que haya más de una se separarán por el símbolo \. La sección está delimitada por los tags <ACCIONES> y <\ACCIONES>.
- **Código Aparato:** Será el código que identifique de forma unívoca al aparato, mediante este código el servidor identificará el aparato con el que se está operando. La sección está delimitada por <CODIGO> y <\CODIGO>.

Tanto los códigos de las acciones como el código del aparato aceptan valores alfanuméricos de longitud variable, la única limitación será la de la clase String de Java, que deberá poder tratar el conjunto del código.

A continuación se muestra el esquema, teniendo en cuenta que lo que no esté de color rojo debe considerarse literales que deben estar presentes:

```
<TITULO>[Nombre Aparato]<\TITULO>  
<ACCIONES>[acción 1>::[cód. op 1][acción 2>::[cód. op 2][acción n>::[cód. op n]<\ACCIONES>  
<CODIGO>[Código Aparato]<\CODIGO>
```

Ilustración 41: Esquema de marcador de códigos QR

A continuación se muestra un ejemplo de código reconocido por la aplicación:

```
<TITULO>Radio<\TITULO>  
<ACCIONES>apagar::0\encender::1\bajar volumen::2\subir volumen::3<\ACCIONES>  
<CODIGO>0003<\CODIGO>
```

Ilustración 42: Ejemplo de marcador de código QR

## **ANEXO III– Acrónimos, definiciones y referencias.**

A continuación se van detallar los acrónimos aparecidos en el documento así como los conceptos que puedan ser de dudoso entendimiento, para ello se definirá su significado en relación a la temática de este proyecto.

### **Acrónimos**

**ADT** – Android Development Tool

**AENOR** – Asociación Española de Normalización y Certificación

**API** – Application Programming Interface

**AR** – Aumented Reality

**ART** – Android RunTime

**BACS** – Building Automation and Control System

**CEM** – Compatibilidad Electromagnética

**CUDA**– Compute Unified Device Architecture

**FPS** – Frames per Second

**GPL** – General Public License

**HAL** – Hardware Abstraction Layer

**HBES** – Home and Building Electronic Systems

**ICT** – Infraestructuras Comunes de Telecomunicaciones

**IDC** – Internet Data Corporation

**IPC** – Inter-Process Communication

**ITC** – Instrucciones Técnicas Complementarias

**IP** – Internet Protocol

**SO** – Sistema operativo

**SDK** – Software Development Kit

**QR Code** – Quick Response Code

**REBT** – Reglamento Electrotécnico para Baja Tensión

**UML** – Unified Modeling Language

**UNE-EN** – Una Norma Española – European Norm

**XML** – Extensible Markup Language

## Definiciones

**Depuración** - Término que hace referencia a la acción de analizar el funcionamiento de algo con el objetivo de verificar su funcionamiento.

**Driver** – Término que hace referencia a un programa informático que permite al sistema operativo interactuar con dispositivo.

**Fork** – Término que hace referencia a la creación de un proyecto en una dirección distinta a la principal u oficial, tomando el código fuente del proyecto ya existente.

**Framework** – Término que hace referencia a una estructura conceptual y tecnológica de soporte definido, que puede servir de base para la organización y desarrollo de software.

**Google Play** – Hace referencia a la tienda virtual de Google, donde se puede adquirir contenido digital y dispositivos tecnológicos.

**Material Desing** – Hace referencia al estilo propuesto por Google con el lanzamiento de la versión 5.0 de Android.

**Kernel** – Término que hace referencia a la parte fundamental del sistema operativo.

**Intent** – Término que hace referencia a una descripción abstracta de una operación a realizar.

**Layout** – Término utilizado para hacer referencia a la capa de visualización.

**Plugin** – Término que hace referencia a un complemento o añadido que permite extender funciones.

**Runtime** – Término que hace referencia al tiempo de ejecución.

## Referencias

1. Asociación Española de Domotica e Inmótica2015, . Available: <http://www.cedom.es/es> [2015, 2015-07-19].
2. Estadísticas de Android2015, 2015-08-03-last update. Available: <https://developer.android.com/about/dashboards/index.html> [2015, 2015-08-16].
3. Página de DENSO, creadores de los códigos QR2015, . Available: <http://www.globaldenso.com/en/>.
4. Página oficial de Samsung2015, . Available: <http://www.samsung.com/global/business/semiconductor/minisite/Exynos/w/index.html> [2015, 2015-07-26].
5. Página web del procesador Helios X20 de Mediatek2015, . Available: <http://heliox20.com/> [2015, 2015-27-26].
6. Página web del proyecto ZXing2015, 2015-02-15-last update. Available: <http://zxing.github.io/zxing/> [2015, 2015-07-18].
7. Procesadores Qualcomm2015, . Available: <https://www.qualcomm.com/products/snapdragon/processors/comparison> [2015, 2015-07-26].
8. Android API2014, 2014-12-19-last update. Available: <http://developer.android.com/guide/index.html> [2014, 21-12-2014].
9. Generador códigos QR2014, 2014-09-07-last update. Available: <http://www.codigos-qr.com/generador-de-codigos-qr/> [2014, 2014-12-21].
10. OpenCV2014, 2014-11-19-last update. Available: <http://opencv.org/> [2014, Diciembre 2014].
11. WEMO2014, . Available: <http://www.belkin.com/us/Products/home-automation/c/wemo-home-automation/> [2014, 2014-12-21].
12. NyARToolkit2013, 2013-11-21-last update. Available: <http://nyatla.jp/nyartoolkit/wp/> [2014, 2014-12-2014].
13. Open Handset Alliance2011, 2011-07-18-last update. Available: <http://www.openhandsetalliance.com/> [2015, 2015-08-16].
14. Proyecto AndAR2010, 2010-11-29-last update. Available: <https://code.google.com/p/andar/> [2014, 2014-12-21].
15. Página creada por DENSO WAVE con información acerca de los códigos QR. Available: <http://www.qrcode.com> [2015, .
16. Ayuntamiento de Madrid 2014, 2014-09-19-last update, Programa piloto 'Código QR' para enfermos de Alzheimer. Available: <http://www.madrid.es/portales/munimadrid/es/Inicio/Ayuntamiento/Medios-de-Comunicacion/Notas-de-prensa/Programa-piloto-Codigo-QR-para-enfermos-de-Alzheimer?vgnextfmt=default&vgnextoid=86211f4865d88410VgnVCM2000000c205a0aRCRD&vgnnextchannel=6091317d3d2a7010VgnVCM100000dc0ca8c0RCRD> [2014, 2014-12-21].
17. International Data Corporation (IDC) 2014, Q3 2014-last update, Smartphone OS Market Share, Q3 2014. Available: <http://www.idc.com/prodserv/smartphone-os-market-share.jsp> [2015, 05-01-2015].
18. Palacio, J. 2015, 2015, Abril-last update, Libro Gestión de Proyectos Scrum Manager [Homepage of Scrum Manager®], [Online]. Available: <http://www.scrummanager.net/blog/2015/04/libro-gestion-de-proyectos-con-scrum-manager/> [2015, 2015-08-15].